On the Algorithmic Synthesis of Fluid and Thermofluid Systems

DISSERTATION
zur Erlangung des Grades eines Doktors
der Ingenieurwissenschaften

vorgelegt von
M.Sc. Jonas Benjamin Weber

eingereicht bei der Naturwissenschaftlich-Technischen Fakultät
der Universität Siegen
Siegen 2021

Betreuer und erster Gutachter
Prof. Dr. rer. nat. Ulf Lorenz
Universität Siegen

Zweiter Gutachter
Prof. Dr.-Ing. Peter F. Pelz
Technische Universität Darmstadt

Tag der mündlichen Prüfung
16. September 2021

Danksagung

An dieser Stelle möchte ich allen danken, die zum Gelingen dieser Arbeit beigetragen haben.

Mein besonderer Dank hierbei gilt Herrn Prof. Dr. Ulf Lorenz, der diese Arbeit betreut hat und mir bei der Durchführung stets die vollste Unterstützung zukommen ließ. Insbesondere der konstruktive Austausch und die regelmäßigen Gespräche auf fachlicher und persönlicher Ebene waren stets eine große Hilfe für mich und haben mich positiv beeinflusst und ermutigt.

Bei Herrn Prof. Dr.-Ing. Peter Pelz bedanke ich mich für die Begutachtung dieser Arbeit und die wissenschaftliche Zusammenarbeit in den letzten Jahren.

Großer Dank gilt auch meinen Kollegen für viele wertvolle Anregungen und konstruktive Kritik, besonders aber für das freundschaftliche Arbeitsklima.

Nicht zuletzt gilt ganz besonderer Dank meiner Familie und meiner Partnerin, die mir stets Mut zugesprochen und mich in meiner Arbeit bestärkt haben. Ich möchte ihnen an dieser Stelle von ganzem Herzen für ihre bedingungslose Unterstützung danken.

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides statt, dass ich die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer, nicht angegebener Hilfsmittel angefertigt habe. Die aus anderen Quellen direkt oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet.

Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form einer anderen Prüfungsbehörde vorgelegt.

Es wurden keine Dienste eines Promotionsvermittlungsinstituts oder einer ähnlichen Organisation in Anspruch genommen.

Siegen, 07. Mai 2021

Zusammenfassung

Einzelne technische Komponenten sind in der Regel bereits sehr weit ausgereift. Der Entwurfsprozess ganzer technischer Systeme ist jedoch, insbesondere in seinen frühen Phasen, immer noch stark von der praktischen Erfahrung von Anlagenplanern geprägt. Gerade das Zusammenspiel der einzelnen Systemkomponenten unter variierenden Bedingungen führt zu einer hohen Entwurfskomplexität. Die Anwendung moderner algorithmischer Optimierungsverfahren bietet hierbei das Potenzial, Kosten zu senken und gleichzeitig die Energieeffizienz zu steigern. Die übergeordnete Vision ist daher der verstärkte Einsatz dieser Methoden zur Unterstützung des Entwurfsprozesses technischer Systeme. Als Beitrag hierzu wird die algorithmische Systemsynthese von Fluid- und Thermofluidsystemen untersucht. Ziel ist es, aus verfügbaren Optionen einzelne Komponenten auszuwählen und zu kombinieren, um ein optimales Gesamtsystem zu erhalten, das in der Lage ist, ein definiertes Kollektiv unterschiedlicher Lasten mit minimalen Kosten abzudecken.

Ausgangspunkt hierfür ist die Vorstellung eines Optimierungsmodells für die Synthese von Fluidsystemen. Darauf aufbauend werden Modellerweiterungen und algorithmische Methoden entwickelt, die auf eine optimale Synthese von Systemen im praxisrelevanten Maßstab abzielen. Für Fluidsysteme wird ein Verfahren vorgeschlagen, das anwendungsspezifische Eigenschaften ausnutzt und in der Lage ist, die Leistungsfähigkeit kommerzieller Standardsoftware zu übertreffen. Hierbei wird der Simulated Annealing Algorithmus verwendet, um gute Anfangslösungen für das Verfahren zu generieren, woraufhin ein anwendungsspezifisches Relaxationsverfahren vorgestellt wird, um starke Schranken für die Bewertung der gefundenen Lösungen zu identifizieren. Im Anschluss können dann mit Hilfe der Branch-and-Bound Methode global optimale Lösungen ermittelt werden. Um die entwickelte Methodik zu testen, werden Druckerhöhungsanlagen als Anwendungsbeispiel herangezogen. Weiterhin wird ein Ansatz gezeigt, wie die Resilienz der erzeugten Systeme in Hinsicht auf den Ausfall einzelner Komponenten erhöht werden kann.

Außerdem wird eine mit dem Fluidsystemmodell kompatible Modellerweiterung für die Betrachtung von Thermofluidsystemen vorgestellt. Während diese Betrachtung dynamisches Verhalten ausklammert, das sich aus der Kopplung benachbarter Zeitpunkte durch Speicherkomponenten ergibt, werden daran anschließend zwei geeignete Ansätze hierfür vorgestellt. Dabei handelt es sich zum einen um einen traditionellen, zeitdiskreten Ansatz und zum anderen um einen neuartigen, zeitkontinuierlichen Ansatz, der auf der Betrachtung variabler Zeitintervalle basiert.

Die vorliegende Arbeit soll somit einen Ausgangspunkt für weitere Forschung auf dem Gebiet der algorithmischen Synthese technischer Systeme und deren Transfer in die Praxis bilden.

Abstract

Individual technical components are usually well optimized. However, the design process of entire technical systems, particularly in its early stages, is still dominated by the practical experience of system designers. Especially the interaction of the individual system components at varying conditions leads to a high level of design complexity. In this context, the application of modern algorithmic optimization methods offers the potential to reduce costs and at the same time increase energy efficiency. The overarching vision is therefore the widespread use of these methods to support the human-driven design process of technical systems. As a contribution to this, the algorithmic system synthesis of fluid and thermofluid systems using mathematical programming techniques is examined in this thesis. For this task, the objective is to select and combine individual components with defined characteristics from a set of available options in order to obtain an optimal overall system capable of covering a given collective of varying loads at minimum cost.

The starting point is the presentation of an optimization model for the synthesis of fluid systems. Based on this, model extensions and algorithmic methods are developed that aim at an optimal synthesis of systems on a practice-oriented scale. For fluid systems, an approach is proposed that exploits domain-specific properties and is able to outperform commercial standard solvers. Simulated Annealing is used to generate good initial solutions for the approach, whereupon a domain-specific relaxation procedure is used to identify strong bounds for evaluating the obtained solutions. In a subsequent step, globally optimal solutions can be obtained using the Branch-and-Bound method. This approach is then further analyzed for the application example of booster stations. Furthermore, it is demonstrated how the resilience of existing systems can be increased with respect to the failure of individual components by using quantified programming.

Moreover, a model extension is presented to extend the consideration to thermofluid systems. These systems are assumed to comprise fluid systems with superimposed heat transfer, which ensures compatibility with the basic fluid system model. However, while the previous considerations exclude dynamic behavior that results, inter alia, from the coupling of adjacent points in time by storage components, two time representations suitable for this application are presented and discussed. One of these two representations is a more traditional discrete-time approach and the other is a novel continuous-time approach based on the consideration of variable time intervals.

Overall, this thesis is intended to provide a foundation for further research in the field of algorithmic synthesis of technical systems and its transfer to engineering practice.

Contents

Physical Quantities

In the following, the most relevant physical quantities for this thesis are introduced. The associated symbols are listed in the first column and described in the second column. In the third column, the corresponding dimension of each quantity is given using the basic quantities mass (M), length (L), time (T), and temperature (K). It should be noted that the symbols introduced here apply only in the context of general physical notations. In other contexts, especially in the case of optimization models, the symbols may differ and the specific notation is given explicitly for the particular domain under consideration.

Symbol	**Description**	**Dimension**
A	Area	L^2
c	Specific heat capacity	$LT^{-2}K^{-1}$
E	Energy	ML^2T^{-2}
g	Acceleration due to gravity	LT^{-2}
h, H	Head	L
i	Specific enthalpy	L^2T^{-2}
m	Mass	M
$\dot{m}$	Mass flow rate	MT^{-1}
ν	Specific volume	L^3M^{-1}
p	Pressure	$ML^{-1}T^{-2}$
P	Power	ML^2T^{-3}
Q	Heat	ML^2T^{-2}
$\dot{Q}$	Heat flow	ML^2T^{-3}
ρ	Density	ML^{-3}
t, τ	Time	T
T	Temperature	K
U	Internal energy	ML^2T^{-2}
u_t	Specific thermal energy	L^2T^{-2}
U_t	Thermal energy	ML^2T^{-2}
$\dot{U}_t$	Thermal energy flow rate	ML^2T^{-3}
v	Velocity	LT^{-1}
V	Volume	L^3
$\dot{V}$	Volume flow rate	L^3T^{-1}
W	Work	ML^2T^{-2}
z	Height	L

Acronyms

ACC	Aggregated Convex Combination method
B&B	Branch-and-Bound method
COP	Coefficient of Performance
DEP	Deterministic Equivalent Program
DLog	Logarithmic Disaggregated Convex Combination method
DP	Dynamic Programming
EEI	Energy Efficiency Index
EER	Energy Efficiency Ratio
EU	European Union
FEM	Finite Element Method
GACC	Generalized Aggregated Convex Combination method
GUI	Graphical User Interface
HVAC	Heating, Ventilation and Air Conditioning
INC	Incremental method
IP	Integer Program
LB	Lower Bound
LP	Linear Program
MILP	Mixed-Integer Linear Program
MINLP	Mixed-Integer Nonlinear Program
MPC	Model Predictive Control
NLP	Nonlinear Program
OR	Operations Research
QMIP	Quantified Mixed-Integer Program
SA	Simulated Annealing algorithm
SOS	Special Ordered Set
TES	Thermal Energy Storage
TOR	Technical Operations Research
UB	Upper Bound

List of Figures

List of Tables

1 Introduction

> What I cannot create, I do not understand.
>
> Richard P. Feynman

1.1 Motivation

At the COP21 climate conference in Paris, an agreement to mitigate climate change and take action to achieve a sustainable low-carbon future was reached. The goal of this agreement is to keep the global temperature increase to “well below” 2°C and to aim for a temperature increase of less than 1.5°C (UNFCCC, 2016). Alongside the expansion of renewable energies, focusing on enhanced energy efficiency is a key aspect for achieving the greenhouse gas reduction goals defined by the participating parties (IRENA, 2019). In this spirit, the German Federal Ministry for Economic Affairs and Energy proclaimed in a recent article on energy efficiency that “the cleanest and cheapest energy is energy that we don’t use in the first place. In order to make our energy transition a success, we need to become even more efficient in the way we use power and heat” (BMWi, 2019). Besides this political level, the respective end users of energy, especially the industrial sector, should also have a personal interest in increasing energy efficiency for economic reasons or as the German Federal Ministry for Economic Affairs and Energy puts it “energy efficiency equates to cost efficiency—a clear competitive advantage” (BMWi, 2010). The ministry further states that in industry, investments in enhanced energy efficiency oftentimes have payback periods of less than two years and yield a return of investment of 25% or more (BMWi, 2010). This underlines that the economic efficiency of an investment should not only be evaluated on the basis of the initial expenditure but also with regard to the entire life cycle costs, including energy costs as well as maintenance, downtime, and disposal costs.

In light of this discussion, two closely related sectors, both of which have a high potential for significant savings, can be highlighted. On the one hand, there is the pumping sector. The yearly energy consumption of all electric pumps in the European Union (EU) is estimated at around 300 TWh, which corresponds to approximately 10% of its net electricity generation in 2018 (Betz, 2017; Eurostat, 2020; VDMA, 2019). The relevance of pumps in terms of energy efficiency is also demonstrated by the fact that, when looking for products and product groups with both high energy consumption and significant savings potential, the European Commission considered pumps as candidates for regulation already at an early stage of the selection process and as a result commissioned a detailed study to uncover the full potential (VDMA, 2019). On the other hand, there is the heating and cooling sector. According to the European Commission (2016), this sector consumed 50% (546 Mtoe[1]) of the EU's final energy in 2012, making it the largest energy sector. The Commission further states that a large portion of this energy is wasted, which is especially alarming when considering that 75% of primary energy still comes from fossil fuels. According to the Commission, all three consumer groups, i.e. private households, industry, and services, yield potential for reducing demand and increasing efficiency. As can be seen, both sectors individually already provide a high savings potential. However, paying attention to the intersection of both areas uncovers even further energy-saving potential since pumps are used for distribution or circulation in heating and cooling systems and the groups of components involved in these applications are closely linked.

Nevertheless, only focusing on the individual components is not enough. In practice, individual technical components are nowadays already typically well optimized in terms of energy efficiency. However, Müller et al. (2020) note that although efficient components are a prerequisite for an efficient overall system, their utilization does not necessarily result in an efficient system. This consideration and the associated decisions are highly relevant to the final system and its efficient operation. Empirical studies suggest that the initial decisions, i.e. combining the intended functionality, layout and used components as well as the expected loads for the future use, make up 70–85% of a system's total life cycle costs (VDI 2884). While awareness is generally increasing in this regard, the practical implications remain mostly unclear. This is especially critical since the savings potential resulting from the consideration of individual components as a system with interdependencies and varying load conditions is, however, also opposed by a higher number of possible

[1] Mega tonnes of oil equivalent, 1 Mtoe = 11.63 TWh

influencing factors that have to be considered and thus a higher planning complexity. With regard to pump systems, for example, there is an international standard ISO/ASME 14414 that deals with the system-wide energy assessment, but according to Müller et al. (2020), the recommendations proposed therein do not include statements on the practical design process and validation of the envisaged energy savings.

The detailed design process of arbitrary technical products and systems is hard to formalize in a general way since it varies depending on the industry, application under consideration and specific requirements. However, guidelines such as VDI 2221 try to identify a universally applicable, basic approach. For this, a classification of the planning process into four distinct phases can be made: clarification and specification of the task, conceptual design, embodiment design, and detail design (Pahl et al., 2007). VDI 2221 discusses these planning phases in general terms, while additional guidelines go into more detail about specific phases. In this regard, VDI 2222 comprises a detailed examination of the first two phases and VDI 2223 deals with the third phase but focuses on applications related to mechanical engineering and precision mechanics in particular. However, it is explicitly stated that other application areas such as plant engineering, process engineering or software development are not excluded and that the statements made may also be applied to these as well. Nevertheless, in the context of these guidelines, it becomes clear that the design process is still dominated by tacit knowledge and the practical experience of the individual designers. To name only a few examples, VDI 2223 states that due to the "solution explosion" of the number of system variants, it is mostly impossible to generate and examine all possible variants even for simple cases. Therefore, the importance of experience is highlighted in the guideline and it refers to rules that ensure a "probably correct procedure". Another example is that so-called design critique, i.e. the consultation of experienced colleagues in order to increase the reliability of the evaluation of possible designs, is pointed out as an important tool.

While in other areas of engineering sciences, such as production planning and logistics, algorithmic optimization techniques—often summarized under the term "Operations Research"—are already widely used in practice and have been providing efficient support to decision makers for quite some time, the adaptation of these methods for technical applications has only recently received increased attention but is still rarely explored and barely used (Dörig et al., 2014). Although today solid results in the context of the application of mathematical programming techniques have been produced for example regarding the optimization of water (see e.g. Martin et al., 2012) and gas networks (see e.g. Domschke et al., 2011), the transfer to operational practice has not yet taken place on a large scale. One reason could be that most studies have in common that mathematicians and designers have chosen a particular subtask with the aim of designing a specific technical system that has been thoroughly examined with respect to special characteristics in the problem structure and solved using advanced, specialized algorithms. Two key elements can be identified from this. On the one hand, it seems promising to make the underlying methodology accessible to a broader audience instead of selectively focusing on a particular application within a specific project. On the other hand, building a bridge between the two disciplines, promoting acceptance and getting both disciplines on board is of utmost relevance for success. Establishing the widespread application is only possible through the interaction of both disciplines. The expertise in providing adaptable optimization models and proposing suitable algorithmic methods is as important as the designers' detailed knowledge regarding the design process as well as the relevant technical background in connection with the assumptions and simplifications to be made in the models, which enables a sound interpretation of system proposals.

1.2 Own Contribution to Knowledge

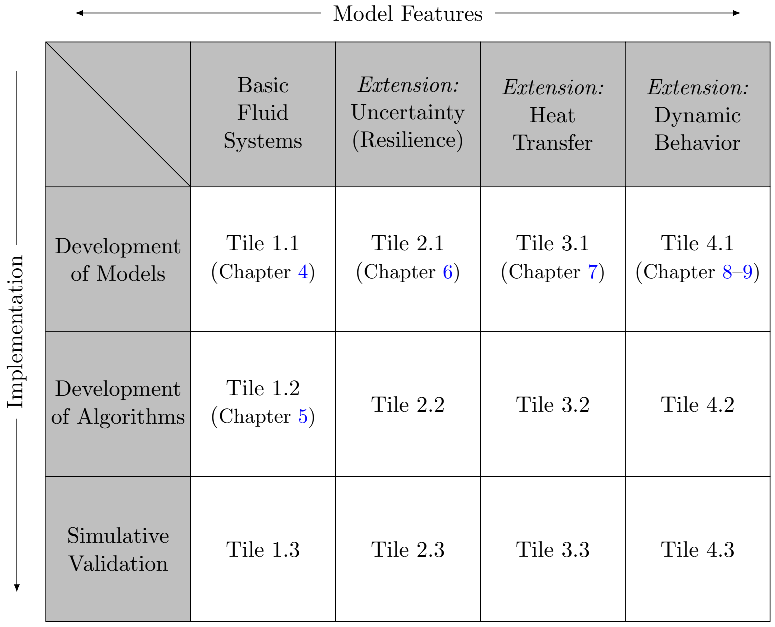
This thesis addresses the optimization of fluid and thermofluid systems with a distinct focus on providing decision support to system designers through the use of quantitative, modern algorithmic methods. In this regard, the scope of the contributions presented in this work focuses on the consideration of incompressible fluids. For the sake of simplicity, all models and methods are stated for the application case of water-based systems. Nevertheless, the adaptation for other incompressible fluids is straightforward, provided that the individual physical properties of the respective medium are taken into account.

In the context of this work, an attempt is made to combine the strengths of different approaches that are applied for the optimization of technical systems. These approaches are, on the one hand, the application of mathematical programming techniques and, on the other hand, the use of simulations. One strength of mathematical programming techniques over simulation-based approaches is that rather than simulating and iteratively optimizing the parameters of one particular, largely predetermined system proposal, a suitable system proposal is generated by simultaneously considering a large number of implicitly described systems. Furthermore, and in contrast to purely heuristic approaches, which are often seen in the context of simulation-based optimization, the mathematical programming techniques discussed in this thesis guarantee global optimality within the model and thus provide an absolute benchmark for the evaluation of the generated system proposal. However, the application of mathematical programming techniques often focuses on solving one specific problem, whereas a strength of simulation-based approaches is the fact that the simulation environments used are able to examine a wider variety of different problems, typically using a more generic representation with reusable and customizable templates. In this context, the goal is not to identify and solve one specific engineering problem but to provide designers with the means to define their individual problems formally in a way that is necessary for the application of mathematical programming techniques and to support them during the solution process. This also includes being guided to formulate tasks within their own established technical language and not having to go into too much detail about the respective methodology.

Accordingly, the major challenge is to be able to model the design of fluid and thermofluid systems in a general and consistent way similar to the widespread simulation environments such as Openmodelica[2] or Matlab/SimuLink[3] and at the same time to be able to perform algorithmic optimization. The focus is that, due to a modular principle, system designers should have the possibility to pick out relevant elements for their application and extend or modify them if necessary. All elements should be based on the same foundation, as it is common for the above mentioned simulation environments. With this in mind, however, the development of suitable models and methods for the design of general fluid-based systems to a practical extent is a visionary challenge. Therefore, the decomposition into sub-challenges, as shown in Figure 1.1, is necessary. Starting in the upper left corner with the basic fluid system model, the investigation can unfold in two different dimensions. The first dimension is the extension of the fluid system model in order to include additional features. This comprises the consideration of uncertainty, in particular resilience, heat transfer as well as dealing with dynamic system behavior, e.g. caused by storage components. The second dimension is the degree of implementation, from the formulation of suitable models and model extensions for which instances can be solved on a laboratory scale with the help of standard solvers,

[2] https://www.openmodelica.org (accessed May 07, 2021)
[3] https://mathworks.com/products/simulink.html (accessed May 07, 2021)

to the development of sophisticated heuristics and exact algorithms exploiting the domain-specific but still general features for handling larger instances, to the validation of proposed solutions by means of detailed simulation.


Model Features
Implementation

	Basic Fluid Systems	*Extension:* Uncertainty (Resilience)	*Extension:* Heat Transfer	*Extension:* Dynamic Behavior
Development of Models	Tile 1.1 (Chapter 4)	Tile 2.1 (Chapter 6)	Tile 3.1 (Chapter 7)	Tile 4.1 (Chapter 8–9)
Development of Algorithms	Tile 1.2 (Chapter 5)	Tile 2.2	Tile 3.2	Tile 4.2
Simulative Validation	Tile 1.3	Tile 2.3	Tile 3.3	Tile 4.3

Figure 1.1: Overview of the research sub-challenges and the contributions in this thesis

The sub-challenges examined in this work are highlighted in Figure 1.1 by tiles indicating a chapter and form the foundation for further research in this area. From these sub-challenges, five key research questions emerge:

- **Tile 1.1.** How can a suitable basic model for fluid systems be represented that can serve as a starting point for further investigations?
- **Tile 1.2.** How can domain-specific engineering knowledge be meaningfully integrated into the solution process to solve larger, practice-oriented instances compared to using standard solvers?
- **Tile 2.1.** How can the approach be extended to include the consideration of resilience as an additional technical requirement, if necessary?
- **Tile 3.1.** How can the basic model for fluid systems be extended in order to allow the consideration of thermofluid systems and thus heating and cooling applications?

- **Tile 4.1.** How can the dynamic behavior that occurs, inter alia, due to energy storage in heating and cooling applications be accounted for and appropriately integrated into the existing modeling approach?

In order to address the research questions defined above and provide significant progress towards the overall vision, the contributions of this thesis are:

- **Tile 1.1.** A modular basic model for fluid systems is presented. Due to the modular structure, the model can easily be extended to include new features. Accompanying this, the strict separation of system and component behavior allows new components to be added easily or existing component descriptions to be adapted as required for the particular application.

- **Tile 1.2.** Domain-specific solution methods for solving instances that exceed the capability of standard solvers are presented and analyzed in an extensive computational study with specially developed test instances using the application example of so-called booster stations for the drinking water supply of high-rise buildings. The successful application of the underlying domain-specific engineering knowledge underlines the potential of an interdisciplinary perspective. In this context, an adapted implementation of the Simulated Annealing algorithm is used to obtain primal solutions. The implementation uses a graph-based representation of fluid systems in combination with the theory of so-called series-parallel networks. In order to evaluate the heuristically obtained solutions, a domain-specific relaxation is developed. The relaxation is based on the decomposition of the overall problem by exploiting implicit knowledge to generate tight dual bounds for the considered problem. Furthermore, it is shown how both methods can be combined in a Branch-and-Bound framework to close the gap between primal and dual solutions and obtain globally optimal solutions.

- **Tile 2.1.** A quantified programming-based approach to increase the resilience of technical systems as a subsequent design decision is presented. The approach, which can be applied to a broad range of applications, is demonstrated using the application example of resilient booster stations that are guaranteed to maintain operation in the event of component failure, with the additional requirement that each component must be operational at least once per day. Furthermore, different potential purposes of use are highlighted.

- **Tile 3.1.** For the consideration of thermofluid systems, a model extension is proposed that allows the inclusion of heating and cooling applications. The model for fluid systems remains valid, unaffected by this extension, since it is designed in such a way that the thermal properties can be superimposed without having to modify the basic model. In addition, the model extension is aligned with the modular structure of the basic model so that the combined model can also easily be extended and adapted.
- **Tile 4.1.** While the previous models and extensions consider only a sequence of different system states without temporal coupling of time steps, the concept of time-dependent behavior of a system, using the example of storage components, is outlined. Two alternative approaches are presented and examined, a discrete-time representation and a novel continuous-time representation with variable time step lengths. Both approaches are compared and their strengths and weaknesses are discussed. While discrete-time representations are widely used, continuous-time representations are used rather rarely and require further consideration. The extent to which their application and the associated limitations and trade-offs are reasonable can only be decided depending on the technical application and its requirements, which in turn emphasizes the need for an interdisciplinary perspective.

The research presented in this thesis is based in part on the following publications, with work done in close collaboration with the respective co-authors:

- J. B. Weber and U. Lorenz. Optimizing booster stations. In *GECCO '17: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1303–1310, New York, 2017. ACM
- M. Hartisch, A. Herbst, U. Lorenz, and J. B. Weber. Towards resilient process networks—designing booster stations via quantified programming. In *Uncertainty in Mechanical Engineering III*, pages 199–210, Switzerland, 2018. Trans Tech Publications Ltd
- J. B. Weber and U. Lorenz. Algorithmic system design of thermofluid systems. In *EngOpt 2018 Proceedings of the 6th International Conference on Engineering Optimization*, pages 132–143, Cham, 2019a. Springer
- J. B. Weber and U. Lorenz. Modeling thermofluid systems: An approach customized for optimization. In *Operations Research Proceedings 2018*, pages 387–393, Cham, 2019b. Springer

– J. B. Weber, M. Hartisch, A. Herbst, and U. Lorenz. Towards an algorithmic synthesis of thermofluid systems. *Optim Eng*, 2020a. doi: https://doi.org/10.1007/s11081-020-09564-1

– J. B. Weber, M. Hartisch, and U. Lorenz. Optimized design of thermofluid systems using the example of mold cooling in injection molding. In *Operations Research Proceedings 2019*, pages 473–480, Cham, 2020b. Springer

1.3 Structure of the Thesis

This introduction is followed by nine chapters, organized as follows:

– **Chapter 2: Theoretical Background.** In this chapter, an overview of the underlying theoretical principles is given. First, the basic engineering and physical equations of fluid mechanics and heat transfer most relevant to this thesis are presented in detail. Then, the most important terms and concepts related to linear programming are introduced. In addition, it is shown how univariate and multivariate nonlinearities can be approximated piecewise linearly in order to be able to apply linear programming techniques. Furthermore, the basic structure as well as the properties of the considered system synthesis problems are discussed.

– **Chapter 3: Optimization-Supported System Design.** To start the chapter, a systematic design approach, which aims at complementing the conventional system design procedure by algorithmic optimization, is presented and discussed. Afterwards, the concepts, ideas and results of related work as well as adjacent research areas are exemplified.

– **Chapter 4: Modeling of Fluid Systems.** The chapter opens with a general description of fluid systems and the introduction of the problem statement. Additionally, it is shown how the considered problem can be alternatively represented as a mathematical graph. Subsequently, the technical component types of fluid systems relevant for this thesis are introduced. Based on this, the basic model for fluid systems is presented, which forms the starting point for further investigations.

– **Chapter 5: Algorithmics for the System Design of Fluid Systems.** At the beginning of the chapter, the design of test instances for the considered application example of booster stations for supplying high-rise buildings with drinking water is discussed. Afterwards, an adapted implementation of the Simulated Annealing algorithm and a domain-specific relaxation are presented, which are used to obtain good initial solutions and to be able to evaluate them. These are then further used in a Branch-and-Bound framework to obtain globally optimal solutions. Finally, an extensive computational study is conducted to test the developed methods using the previously discussed test instances. For this, the obtained solutions are examined in terms of solution quality, system topology, and runtime.

- **Chapter 6: Resilient System Design.** Starting with an outline of the concept of quantified programming, an optimization model for increasing the resilience of technical systems is presented using resilient booster stations as an application example. The approach is then tested for two test instances that illustrate the different possible uses of the approach and the results are discussed.

- **Chapter 7: Modeling of Thermofluid Systems.** Thermofluid systems and their characteristics are introduced. For these systems, a general classification of the corresponding component types is presented and examples of specific components are described in detail. This is followed by the introduction of the necessary extensions for the basic fluid system model in order to include heat transfer considerations as well as heating and cooling components. The chapter ends with a computational example based on the synthesis of an industrial cooling system.

- **Chapter 8: Discrete-Time Representation for Thermofluid Systems.** First, general approaches to model time are presented. In this context, thermal energy storage is discussed as an important concept for thermofluid systems, which causes a time-dependent system behavior by coupling neighboring points in time, and a model extension to represent the technical behavior of two-layer stratified storage tanks by a discrete-time representation is presented. Using this extension, the application example from the previous chapter is revisited at the end of the chapter.

- **Chapter 9: Continuous-Time Representation for Thermofluid Systems.** As an alternative to the discrete-time representation of the previous chapter, a novel continuous-time representation with variable time step lengths is presented. For this purpose, the advantages and challenges as well as the underlying principles are discussed. After introducing the associated alternative model extension, both approaches are compared using the previously established example of an industrial cooling system.

- **Chapter 10: Conclusion and Outlook.** To conclude this thesis, a summary of the individual findings and an outlook on future research directions are given.

2 Theoretical Background

For the optimization of real-world engineering problems, insights of both adjacent disciplines, engineering and algorithmic optimization, have to be considered. Therefore, the most relevant engineering and algorithmic basics necessary for the examination of the synthesis of technical fluid-based systems are briefly introduced in the following sections. More specific theoretical details are provided in the respective chapters if necessary.

2.1 Engineering Basics

Fluid mechanics and heat transfer form the theoretical basis for the description of fluid and thermofluid systems. The principles most relevant for this work are explained below. For a deeper insight, see e.g. Tritton (1988) and Incropera et al. (2007).

2.1.1 Continuity Equation

The continuity equation is derived from the principle of mass conservation. This subsection follows the explanations of Tritton (1988) and White (2011) with some modifications to the notation.

Given an arbitrary Eulerian control volume V, as shown in Figure 2.1, that is located completely within a fluid and with fluid freely passing its boundaries, let $d\mathbf{S}$ be an element of the surface with its direction being the outward normal, its magnitude being the element's area and $\mathbf{v}$ being the velocity of fluid leaving that element. With ρ being the density of the fluid, the rate of mass loss over the whole surface of V is

$$\int_S \rho \mathbf{v} \cdot d\mathbf{S}. \tag{2.1}$$

Since the above equation is defined for mass loss of V, an increase of the mass of V would have a negative sign. Furthermore, the total mass in volume V is defined as

$$\int_V \rho \, dV. \tag{2.2}$$

By equating Equation (2.1) and the time derivative of Equation (2.2), one obtains

$$\frac{d}{dt}\int_V \rho \, dV = \int_V \frac{\partial \rho}{\partial t} \, dV = -\int_S \rho \mathbf{v} \cdot d\mathbf{S}. \tag{2.3}$$

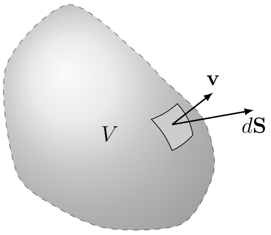


Figure 2.1: Illustration of an arbitrary control volume (based on Tritton, 1988)

For an infinitesimal volume, this results in the general representation of the continuity equations with ρ and $\mathbf{v}$ being depend on the respective position:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{2.4}$$

Assuming steady flow, i.e. $\frac{\partial}{\partial t} = 0$, Equation (2.4) simplifies to

$$\nabla \cdot (\rho \mathbf{v}) = 0. \tag{2.5}$$

Given a control volume with one-dimensional inlets and outlets of cross-sectional area A and assuming steady flow, the following can be stated:

$$\sum_i (\rho_i \cdot v_i \cdot A_i)_{in} = \sum_i (\rho_i \cdot v_i \cdot A_i)_{out} \tag{2.6}$$

Accordingly, for a streamtube along streamlines this can be written as

$$\rho_1 \cdot v_1 \cdot A_1 = \rho_2 \cdot v_2 \cdot A_2. \tag{2.7}$$

If the fluid is incompressible, the equation can be further simplified to

$$v_1 \cdot A_1 = v_2 \cdot A_2. \tag{2.8}$$

An illustration of the continuity equation for fluid flow in a streamtube according to the assumptions applied to Equation (2.8) is shown in Figure 2.2.

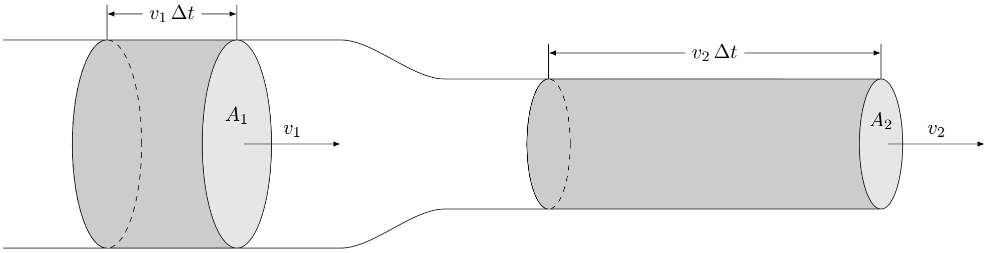


Figure 2.2: Illustration of the continuity equation for fluid flow in a streamtube

2.1.2 Bernoulli's Equation and Losses

The following explanations are based on Munson et al. (2009) and White (2011) with some modifications to the notation. Furthermore, Bernoulli's equation can alternatively be derived by applying Newton's second law instead of using the approach shown here (see e.g. Munson et al., 2009).

Given Euler's equations in vector notation

$$\rho \mathbf{g} - \nabla p = \rho \left[\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} \right], \tag{2.9}$$

with the gravitational acceleration vector $\mathbf{g}$, the Bernoulli equation can be obtained by integrating this equation along an arbitrary streamline. Demanding for steady flow, Equation (2.9) becomes

$$\rho \mathbf{g} - \nabla p = \rho(\mathbf{v} \cdot \nabla)\mathbf{v}. \tag{2.10}$$

The gravitational acceleration vector can also be expressed as

$$\mathbf{g} = -g\nabla z, \tag{2.11}$$

with g being the magnitude of $\mathbf{g}$ and z being the height above a reference plane. Using the vector identity

$$(\mathbf{v} \cdot \nabla)\mathbf{v} = \frac{1}{2}\nabla(\mathbf{v} \cdot \mathbf{v}) - \mathbf{v} \times (\nabla \times \mathbf{v}), \tag{2.12}$$

Equation (2.10) can be rewritten and rearranged (with magnitude $v = ||\mathbf{v}||$) to

$$\frac{\nabla p}{\rho} + \frac{1}{2}\nabla(v^2) + g\nabla z = \mathbf{v} \times (\nabla \times \mathbf{v}). \tag{2.13}$$

With $d\mathbf{s}$ being a differential length along a streamline s, Equation (2.13) becomes

$$\frac{\nabla p}{\rho} \cdot d\mathbf{s} + \frac{1}{2}\nabla(v^2) \cdot d\mathbf{s} + g\nabla z \cdot d\mathbf{s} = \big[\mathbf{v} \times (\nabla \times \mathbf{v})\big] \cdot d\mathbf{s}. \tag{2.14}$$

Along a streamline s, $\big[\mathbf{v} \times (\nabla \times \mathbf{v})\big] \cdot d\mathbf{s} = 0$ because the direction of $d\mathbf{s}$ is always along s and therefore $d\mathbf{s}$ and $\mathbf{v}$ are parallel. Furthermore, using $d\mathbf{s} = dx\,\hat{\mathbf{i}} + dy\,\hat{\mathbf{j}} + dz\,\hat{\mathbf{k}}$, it follows that $\nabla p \cdot d\mathbf{s} = (\partial p/\partial x)\,dx + (\partial p/\partial y)\,dy + (\partial p/\partial z)\,dz = dp$, etc. Hence, with the change in p, v and z being along the streamline, Equation (2.14) yields

$$\frac{dp}{\rho} + \frac{1}{2}\,d(v^2) + g\,dz = 0. \tag{2.15}$$

By integrating, it follows that the sum is constant along a streamline:

$$\int \frac{dp}{\rho} + \frac{v^2}{2} + gz = const. \tag{2.16}$$

For incompressible, inviscid fluids and between two points 1 and 2 along a streamline, as illustrated in Figure 2.3, Equation (2.16) can be expressed as

$$\frac{p_1}{\rho} + \frac{v_1^2}{2} + gz_1 = \frac{p_2}{\rho} + \frac{v_2^2}{2} + gz_2. \tag{2.17}$$

Alternatively, the equation can also be written in its often used head form by dividing each term by g:

$$\frac{p_1}{\rho g} + \frac{v_1^2}{2g} + z_1 = \frac{p_2}{\rho g} + \frac{v_2^2}{2g} + z_2 \tag{2.18}$$

As stated above, Equations (2.17) and (2.18) are in general only applicable to incompressible, inviscid, steady flow along a streamline. However, for flow through pipes, the friction cannot be neglected if long pipe lengths are considered and therefore the assumption of inviscid flow does not hold. Furthermore, the Bernoulli equation cannot be applied if mechanical devices such as pumps and turbines are present along the examined part of a streamline since those are sources and sinks of energy.

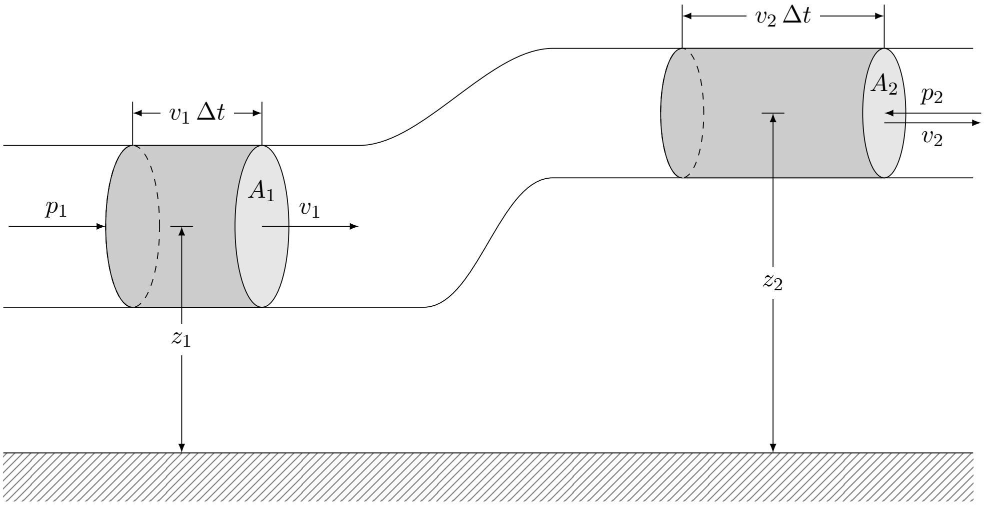


Figure 2.3: Illustration of Bernoulli's equation for fluid flow in a streamtube

In order to include the aforementioned effects resulting in gains and losses of head, Bernoulli's equation can be "modified". This equation, which is actually rather a form of the energy equation (see Section 2.1.3), is called the extended Bernoulli equation

$$\frac{p_1}{\rho g} + \frac{v_1^2}{2g} + z_1 = \frac{p_2}{\rho g} + \frac{v_2^2}{2g} + z_2 + h_L - h_s, \tag{2.19}$$

with h_L being the head loss due to friction between points 1 and 2 and h_s being the head change due to mechanical work on the fluid. For devices increasing the head in flow direction (e.g. pumps and fans), h_s is positive and for devices decreasing the head (e.g. turbines), h_s is negative. Furthermore, only those devices lying between points 1 and 2 are considered. The friction loss h_L is always positive in flow direction. It consists of the so-called major losses $h_{L\,major}$ caused by the friction in straight pipes and the so-called minor losses $h_{L\,minor}$ caused by other components present in a pipe system (e.g. valves, elbows, tees, etc.):

$$h_L = h_{L\,major} + h_{L\,minor} \tag{2.20}$$

The major losses can be expressed by the empirical Darcy-Weisbach equation for incompressible fluids

$$h_{L\,major} = \frac{v^2}{2g}\lambda\frac{l}{d}, \tag{2.21}$$

with pipe length l, (mean) flow velocity v and diameter d (assuming circular cross sections). The friction factor λ is dimensionless and depends on the Reynolds number Re and the relative roughness $\frac{\varepsilon}{d}$:

$$\lambda = \phi\left(Re, \frac{\varepsilon}{d}\right) \tag{2.22}$$

For explanations regarding the Reynolds number and relative roughness, as well as for the determination of the friction factor, see e.g. Munson et al. (2009). The minor head losses can be approximated as

$$h_{L\,minor} = \frac{v^2}{2g}\zeta, \tag{2.23}$$

with loss coefficient ζ. In general, ζ depends on the geometry of the respective component as well as on the Reynolds number, i.e. $\zeta = \phi(geometry, Re)$. However, in most cases of practical interest $\zeta = \phi(geometry)$, which is typically determined either experimentally or by manufacturer's data (Munson et al., 2009). In the case of one pipe with multiple minor losses, Equation (2.20) becomes

$$h_L = h_{L\,major} + \sum h_{L\,minor} = \frac{v^2}{2g}\left(\frac{l}{d}\lambda + \sum \zeta\right). \tag{2.24}$$

2.1.3 Simplified Steady-Flow Thermal Energy Equation

According to Incropera et al. (2007), thermodynamics and heat transfer are related topics. On the one hand, the first law of thermodynamics dealing with the conservation of energy is also an important basic for heat transfer theory. On the other hand, the consideration of the rate of transferred heat can be seen as an extension to thermodynamics. The explanations given in this subsection follow the introduction to heat transfer presented by Incropera et al. (2007).

In general, the first law of thermodynamics for a closed system, i.e. no transfer of matter, states that

$$\Delta E_{St}^{tot} = Q - W, \tag{2.25}$$

with ΔE_{St}^{tot} being the change in total energy stored, Q being the heat into the system and W being the work done by the system. Hence, energy transport across system boundaries occurs as a consequence of heat transfer or work. If it is applied to an open system, e.g. control volume, energy can also be transported by mass, which is called advection.

The total energy can be divided into internal energy and mechanical energy. Internal energy consists of four components, sensible, latent, chemical, and nuclear energy. The sum of sensible and latent energy is called thermal energy. Mechanical energy is the sum of kinetic energy and potential energy. Heat transfer theory typically focuses on thermal and mechanical energy. Therefore, in the following, the sum of thermal and mechanical energy is represented by E. However, only considering thermal and mechanical energy, conservation is not ensured since there may be a conversion between thermal energy and one of the other not considered forms of energy. Hence, it is assumed that thermal energy can be "generated" by the conversion of other energy forms. Thus, the thermal and mechanical energy equation can be written as

$$\Delta E_{St} = E_{in} - E_{out} + E_g. \tag{2.26}$$

ΔE_{st} is the change of the sum of thermal and mechanical energy. E_{in} and E_{out} are the sums of thermal and mechanical energy entering and leaving the system, respectively. E_g is the generation of thermal energy.

On a rate basis, i.e. for an instant t, and using the dot-notation to express rates of physical quantities, Equation (2.26) becomes

$$\dot{E}_{st} = \frac{dE_{st}}{dt} = \dot{E}_{in} - \dot{E}_{out} + \dot{E}_g. \tag{2.27}$$

If advection occurs, i.e. in open systems, the work term is typically divided into the flow work pV due to pressure forces moving the fluid through the boundaries of the control volume and a work term representing the remainder of work $\dot{W}$. For steady-state conditions and without generation of thermal energy, the steady-flow energy equation can be stated as

$$\dot{m}(u_t + p\nu + \frac{1}{2}v^2 + gz)_{in} - \dot{m}(u_t + p\nu + \frac{1}{2}v^2 + gz)_{out} + \dot{Q} - \dot{W} = 0, \tag{2.28}$$

where u_t is the thermal energy, $p\nu$ is the flow work, $0.5v^2$ is the kinetic energy and gz is the potential energy, which are all expressed per unit mass. Multiplying these terms by the rate of mass flow $\dot{m}$ defines the rate at which each energy term enters or leaves the system. However, for many heat transfer applications, the changes in mechanical energy are negligible. Furthermore, the sum of thermal energy per unit mass and flow work per unit mass can also be expressed as specific enthalpy i with

$$i = u_t + p\nu. \tag{2.29}$$

If there are no phase changes, the change of specific enthalpy for an ideal gas with constant specific heat capacities can be written as

$$i_{in} - i_{out} = c_p\,(T_{in} - T_{out}), \tag{2.30}$$

with specific heat capacity at constant pressure c_p and temperatures T_{in} and T_{out}. However, in case of incompressible liquids, as considered in this thesis, the specific heat capacities at constant pressure and at constant volume are the same, i.e. $c_p = c_v = c$. Furthermore, the change in flow work can be neglected if there is no large pressure change. Hence, for incompressible liquids one obtains

$$u_{t,\,in} - u_{t,\,out} = c\,(T_{in} - T_{out}). \tag{2.31}$$

With steady-state conditions, without latent energy changes and thermal energy generation, the simplified steady-flow thermal energy equation

$$\dot{Q} = \dot{m}\,c\,(T_{out} - T_{in}) \tag{2.32}$$

for an incompressible liquid arises if certain conditions are met. The right-hand side represents the net rate of thermal energy leaving the system and the relationship holds if either kinetic and potential energy changes as well as work (including flow work) can be neglected or if viscous dissipation (i.e. viscous forces causing a conversion from mechanical to thermal energy) can be neglected. According to Incropera et al. (2007), most relevant applications meet at least one of these conditions.

2.2 Algorithmic Basics

The optimization of real-world engineering problems typically involves discrete as well as continuous decision variables. Hence, the resulting problems can oftentimes be modeled by Mixed-Integer Programs. Accordingly, this section provides a short overview of Mixed-Integer (Linear) Programming as well as special features regarding the synthesis of technical system. For a deeper insight into the underlying theory of Linear and Integer Programming, see e.g. Schrijver (1986).

2.2.1 Mixed-Integer Linear Programming

A Mixed-Integer Linear Program (MILP) represents an optimization problem with some variables restricted to integer values, while the remaining variables can be real-valued. Besides these restrictions, there may be further constraints that are expressed by linear equations and inequalities. The goal of the optimization is to find a variable assignment that fulfills all constraints as well as minimizes (or maximizes) a linear function, the objective function. The linearity of the objective function and constraints distinguishes a MILP from a Mixed-Integer Nonlinear Program (MINLP). In the case of a minimization problem, a MILP with inequality constraints can be expressed as follows

$$\begin{aligned} & z = \min c^T x \\ s.t. \quad & Ax \leq b \\ & x \in \mathbb{Z}^p \times \mathbb{R}^{n-p}, \end{aligned} \tag{2.33}$$

with decision variable vector $x \in \mathbb{Z}^p \times \mathbb{R}^{n-p}$, cost coefficient vector $c \in \mathbb{Q}^n$, coefficient matrix $A \in \mathbb{Q}^{m \times n}$ and right-hand-side vector $b \in \mathbb{Q}^m$, where p, n, $m \in \mathbb{N}$ and $p \leq n$ (Fügenschuh et al., 2010). MILPs with $p = 0$ are called Linear Programs (LP) and do only have real-valued variables, while those with $p = n$ are called Integer Programs (IP) and are restricted to integer variables exclusively. A special case are MILPs with $p = n$ where $x \in \{0, 1\}$. These are called Binary or 0/1-Programs with variable values restricted to either 0 or 1. Also, a maximization problem can be transformed into a minimization problem by minimizing the negative objective function of the maximization problem and vice versa. Furthermore, equality constraints can be expressed by using two inequality constraints.

In order to define the basic terminology and concepts, the remainder of this subsection follows the definitions given by Nemhauser and Wolsey (1988) with some modifications to the notation: The specification of the data set (c, A, b) yields an "instance" of the optimization problem. The set $S = \{x \in \mathbb{Z}^p \times \mathbb{R}^{n-p}, Ax \leq b\}$ is called the feasible region of an instance and any $x \in S$ represents a feasible solution. A feasible solution $x^* \in S$ that minimizes the value of the objective function, i.e. $c^T x^* \leq c^T x \; \forall \, x \in S$, is called an "optimal solution" and the associated value of the objective function z^* is called the "optimal value". If $S = \emptyset$, there is no feasible solution. Hence, the respective instance is “infeasible”. In contrast, an instance is “unbounded” if for any $\omega \in \mathbb{R}$, there exists a $x \in S$ for which $c^T x < \omega$.

2.2.2 Bounds, Relaxations and Optimality Gap

Given an instance of an optimization problem and some arbitrary feasible solution $x^0 \in S$ with objective value $c^T x^0 = z^0$, it is not known whether this is in fact an optimal solution, i.e. $z^0 = z^*$, or if it is not an optimal solution, how far its objective value deviates from the optimal value. In order to estimate the quality of a solution, lower and upper bounds $\underline{z}$ and $\overline{z}$ for the optimal value z^* can be established such that

$$\underline{z} \leq z^* \leq \overline{z}. \tag{2.34}$$

To explain the associated concept of primal and dual bounds, the following discussion is based on the remarks given by Nemhauser and Wolsey (1988) and Wolsey (1998). On the one hand, the objective value of any feasible solution $x^0 \in S$ represents a primal bound for an optimization problem. In the case of a minimization problem this corresponds to an upper bound with $\overline{z} = c^T x^0$. One way to determine such primal bounds is to use heuristics. On the other hand, dual bounds, which correspond to lower bounds in case of considering a minimization problem, have to be determined. For these, so-called relaxations of the problem are typically used. In general, this means that the original problem is replaced with a simpler problem that is easier to handle but always yields a dual bound for the original problem. A common approach is to obtain relaxations by dropping constraints of the original problem. Furthermore, the term “simpler problem” often means that this problem is within another complexity class than the original problem, e.g. class $\mathcal{P}$ instead of $\mathcal{NP}$. However, this does not necessarily have to be the case and the term is used here in a broader, more intuitive sense. Hence, a problem RP with $z^R = min\{f(x) : x \in T\}$ is said to be a relaxation of the original problem P with $z^P = min\{c(x) : x \in S\}$ if the following two properties hold:

$$S \subseteq T \tag{2.35}$$

$$f(x) \leq c(x) \qquad \forall\, x \in S \tag{2.36}$$

Accordingly, it follows that for the relationship between relaxed problem RP and original problem P the following properties apply:

- If P is feasible, $z^R \leq z^P$.

– If RP is infeasible, P is also infeasible.

– If $x^* \in T$ is an optimal solution of RP and furthermore $x^* \in S$ is a feasible solution of P, then x^* is also an optimal solution of P.

With primal and dual bounds $\overline{z}$ and $\underline{z}$ for a minimization problem, the so-called optimality gap (see e.g. Fügenschuh et al., 2010) can be stated as

$$gap := \frac{\overline{z} - \underline{z}}{\underline{z}}. \tag{2.37}$$

The optimality gap provides information on how far a solution deviates from the optimum. If $gap = 0$ and thus $\underline{z} = \overline{z}$, an optimal solution is found. As long as a gap remains, it is not proven that a solution is optimal and no statement can be made about where the optimum lies in the range between the two bounds. Hence, the optimality gap is only an indicator for the worst-case deviation but not for the actual deviation from the optimum.

2.2.3 Linearization Techniques

The intuitive modeling of optimization problems related to technical applications in many cases involves nonlinear relationships and integer decision variables. However, the resulting MINLPs are in general difficult to solve or even unsolvable (Geißler, 2011). Therefore, the corresponding nonlinear constraints are often piecewise linearly approximated to make them accessible for MILP algorithms. In this thesis, univariate as well as multivariate functions are considered. For both of them, two possible piecewise linear representations are introduced. A detailed description of often used linearization techniques as well as their computational performance is presented in Vielma et al. (2010). The overview given here follows Geißler (2011).

Univariate Functions

In the univariate case, a continuous piecewise linear function $\phi : \mathbb{R} \to \mathbb{R}$ can be approximated using n line-segments (1-simplices) as well as $n+1$ grid points $\overline{x}_i$ ($i \in \{0, ..., n\}$) and their function values $\overline{y}_i = \phi(\overline{x}_i)$ for the domain $[\overline{x}_0, \overline{x}_n]$.

The first technique presented is the Aggregated Convex Combination method (ACC) for univariate functions, sometimes also referred to as λ-method (Vielma et al., 2010). The underlying idea is that function values at a point x can be expressed by a convex combination of grid points, which in the univariate case are the grid points of the line segment on which x lies. Therefore, $n+1$ auxiliary multipliers

$\lambda_0, \dots, \lambda_n \in [0,1]$ (one for each grid point) and n binary variables $z_1, \dots, z_n$ (one for each line segment) are introduced. If one additionally introduces the auxiliary binary variables z_0 and z_{n+1} and fixes them to zero ($z_0 = z_{n+1} = 0$), the model can be stated as follows:

$$x = \sum_{i=0}^{n} \lambda_i \cdot \overline{x}_i \tag{2.38a}$$

$$y = \sum_{i=0}^{n} \lambda_i \cdot \overline{y}_i \tag{2.38b}$$

$$\sum_{i=0}^{n} \lambda_i = 1 \tag{2.38c}$$

$$\sum_{i=1}^{n} z_i = 1 \tag{2.38d}$$

$$\lambda_i \leq z_i + z_{i+1} \qquad \forall\, i \in \{0, ..., n\} \tag{2.38e}$$

$$\lambda_i \geq 0 \qquad \forall\, i \in \{0, ..., n\} \tag{2.38f}$$

$$z_i \in \{0,1\} \qquad \forall\, i \in \{1, ..., n\} \tag{2.38g}$$

The second technique is the Incremental method (INC), also known as δ-method (Vielma et al., 2010). Using this method, a point x in interval i is defined by $x = \overline{x}_{i-1} + (\overline{x}_i - \overline{x}_{i-1}) \cdot \delta_i$ with auxiliary variable $\delta_i \in [0,1]$, where $\delta_i > 0$ only if $\delta_{i-1} = 1$. For this, $n-1$ binary variables $z_1, \dots, z_{n-1}$ are used. Accordingly, the following formulation is obtained:

$$x = \overline{x}_0 + \sum_{i=1}^{n} (\overline{x}_i - \overline{x}_{i-1}) \cdot \delta_i \tag{2.39a}$$

$$y = \overline{y}_0 + \sum_{i=1}^{n} (\overline{y}_i - \overline{y}_{i-1}) \cdot \delta_i \tag{2.39b}$$

$$z_i \leq \delta_i \qquad \forall\, i \in \{1, ..., n-1\} \tag{2.39c}$$

$$\delta_{i+1} \leq z_i \qquad \forall\, i \in \{1, ..., n-1\} \tag{2.39d}$$

$$z_i \in \{0,1\} \qquad \forall\, i \in \{1, ..., n-1\} \tag{2.39e}$$

$$\delta_1 \leq 1 \tag{2.39f}$$

$$\delta_n \geq 0 \tag{2.39g}$$

While ACC is a flexible and widely known technique, Geißler et al. (2012) show that INC performs significantly better in terms of runtime for univariate nonlinearities encountered in the optimization of water supply networks, which have similarities to the systems studied in this thesis. However, due to its widespread use, ACC is considered as an alternative.

Multivariate Functions

Similar to the univariate case, piecewise linear multivariate functions $\phi : \mathcal{D} \subset \mathbb{R}^d \rightarrow \mathbb{R}$ on the compact domain $\mathcal{D}$ partitioned into n simplices (d-simplices) can be approximated using m grid points $\overline{\mathbf{x}}_j$ $(j \in \{1, ..., m\})$. Again, binary auxiliary variables z are introduced to ensure $y = \phi(\mathbf{x})$ holds for $\mathbf{x} \in \mathcal{D}$. For the considered bivariate case, a simplex corresponds to a triangle. To obtain valid triangulations, one can use incrementally constructed Delaunay triangulations (Delaunay, 1934).

The Generalized Aggregated Convex Combination method (GACC) is an extension of ACC to the multivariate case. Similar to the univariate case, any point $\mathbf{x} \in S \subseteq \mathcal{D}$ can be expressed by a convex combination of the vertices $\overline{\mathbf{x}}_j \in \mathcal{V}(S)$ of simplex S. In addition, n binary variables $z_1, \ldots, z_n$ are introduced (one for each simplex). With this, the model can be formulated as follows:

$$\mathbf{x} = \sum_{j=1}^{m} \lambda_j \cdot \overline{\mathbf{x}}_j \tag{2.40a}$$

$$y = \sum_{j=1}^{m} \lambda_j \cdot \overline{y}_j \tag{2.40b}$$

$$\sum_{j=1}^{m} \lambda_j = 1 \tag{2.40c}$$

$$\sum_{i=1}^{n} z_i \leq 1 \tag{2.40d}$$

$$\lambda_j \leq \sum_{\{i | \overline{\mathbf{x}}_j \in \mathcal{V}(S_i)\}} z_i \qquad \forall\, j \in \{1, ..., m\} \tag{2.40e}$$

$$\lambda_j \geq 0 \qquad \forall\, j \in \{1, ..., m\} \tag{2.40f}$$

$$z_i \in \{0, 1\} \qquad \forall\, i \in \{1, ..., n\} \tag{2.40g}$$

While GACC is well established, the Logarithmic Disaggregated Convex Combination method (DLog) is a more recent method proposed by Vielma and Nemhauser (2011) that uses only a logarithmic number of binary auxiliary variables z. In the disaggregated case, a λ-variable is assigned to each of the $d + 1$ vertices of each simplex S out of the set $\mathcal{S}$ of d-simplices instead of one variable per grid point as for the aggregated case. For the logarithmic formulation, the number of z-variables can be reduced from n (one for each simplex) to $\lceil log_2(n) \rceil$. For this, an injective function $c : \mathcal{S} \to \{0,1\}^{\lceil log_2(n) \rceil}$ is introduced. In this context, gray code is used to uniquely encode the respective simplices. There is also a logarithmic formulation of GACC, but this formulation requires a specific triangulation, namely a J_1 triangulation, which is sometimes referred to as "Union-Jack" triangulation (Vielma et al., 2010). However, due to the superior flexibility, DLog is considered instead:

$$\mathbf{x} = \sum_{i=1}^{n} \sum_{j=0}^{d} \lambda_j^{S_i} \cdot \overline{\mathbf{x}}_j^{S_i} \tag{2.41a}$$

$$y = \sum_{i=1}^{n} \sum_{j=0}^{d} \lambda_j^{S_i} \cdot \overline{y}_j^{S_i} \tag{2.41b}$$

$$\sum_{i=1}^{n} \sum_{j=0}^{d} \lambda_j^{S_i} = 1 \tag{2.41c}$$

$$\sum_{i=1}^{n} \sum_{j=0}^{d} c(S_i)_l \cdot \lambda_j^{S_i} \leq z_l \qquad \forall\, l \in \{1, ..., \lceil log_2(n) \rceil\} \tag{2.41d}$$

$$\sum_{i=1}^{n} \sum_{j=0}^{d} (1 - c(S_i)_l) \cdot \lambda_j^{S_i} \leq 1 - z_l \qquad \forall\, l \in \{1, ..., \lceil log_2(n) \rceil\} \tag{2.41e}$$

$$z_l \in \{0,1\} \qquad \forall\, l \in \{1, ..., \lceil log_2(n) \rceil\} \tag{2.41f}$$

$$\lambda_j^{S_i} \geq 0 \qquad \forall\, i \in \{1, ..., n\}, j \in \{0, ..., d\} \tag{2.41g}$$

While most constraints remain unchanged compared to the "conventional" Disaggregated Convex Combination method (see Vielma et al., 2010), Constraints (2.41d) and (2.41e) ensure that only the vertices of the one simplex whose l-th bit of its binary encoding matches z_l for each $l \in \{1, \ldots, \lceil log_2(n) \rceil\}$ are used for the convex combination.

Geißler et al. (2012) show that DLog performs well for applications similar to those considered in this work, namely for multivariate nonlinearities encountered in gas network optimization. However, they also point out that a smaller model does not necessarily lead to shorter runtimes. Therefore and because of its simplicity, GACC is provided as an alternative.

2.2.4 System Synthesis Problems

The synthesis of technical systems, as considered in this thesis, can be viewed as a hierarchical two-stage optimization problem, with binary decision variables representing the purchase decisions to be made for technical components or their operational status, and continuous decision variables representing the operation of the system and the individual components.

Accordingly, a distinction can be made between structural and operational decisions, respectively. While there is only one structural decision phase, there may be multiple operational decision phases equivalent to a certain number of considered, typically discrete, time steps. The structural decisions influence the operational decisions and vice versa. For example, the operation of components associated with the operational decision stage requires that this component is selected for purchase in the hierarchically higher structural stage. Furthermore, it is also possible that the decisions within the operational stage influence each other. Therefore, it seems reasonable to make all decisions of both stages simultaneously in order to exploit the potential that arises from the interaction between and within the two decision stages (Bahl et al., 2018).

Model Structure

According to Bahl et al. (2018), synthesis problems have a special structure, which becomes apparent by examining the respective non-zero coefficient matrices. The situation is illustrated in Figure 2.4. The individual, mutual independent operating blocks $O_1, \ldots, O_{|T|}$ for each time step $t \in T$ may be coupled by "complicating variables" C^V and "complicating constraints" C^C. If there are no complicating variables or complicating constraints, the so-called simple problem arises. This is called the simple problem because it can be simply decomposed into its individual operating blocks, which can be solved independently of each other. However, as in this thesis, it is very common that complicating variables or complicating constraints are present. To enable decomposition in these cases, more advanced approaches such as Benders Decomposition (see e.g. Rahmaniani et al., 2017) or Lagrangean Decomposition (see e.g. Guignard and Kim, 1987) would have to be applied. On the one hand, an example for complicating variables are binary purchase variables that apply to all operating blocks and ensure that all components used for operation must be part of the overall system. On the other hand, examples for complicating constraints are constraints applied to water tanks in order to model the temporal sequence of storage levels, which in turn restricts all operating blocks (Bahl et al., 2018).

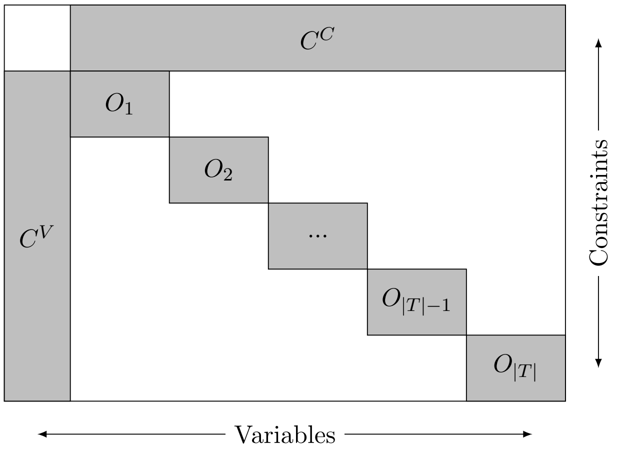


Figure 2.4: Schematic representation of the non-zero coefficient matrix of a synthesis problem with operation blocks $O_1...O_{|T|}$, complicating constraints C^C and complicating variables C^V (based on Bahl, 2018)

Model Complexity

Bahl et al. (2018) further introduce the term “model complexity”, which loosely correlates with the computational effort required. In this context, model complexity should not be confused with the established concept of “computational complexity theory” (see e.g. Schrijver, 1986) and does not imply a strict mathematical relationship. The authors state that the resolution of time and couplings between different time steps are an important factor affecting the model complexity. Accordingly, they identify different influencing factors for increased model complexity:

- Existence of multiple time steps $t \in T$, e.g. caused by different requirements for the volume flow $\dot{V}_t$ at various points in time:

$$\dot{V}_t^{demand} = \dot{V}_t^{supply} \qquad \forall\, t \in T \tag{2.42}$$

- Existence of coupled time steps without the necessity of a chronological time sequence, e.g. caused by the influence of higher-level purchase decisions b on the individual operation decisions a_t for a component in each time step $t \in T$:

$$b \geq a_t \qquad \forall\, t \in T \tag{2.43}$$

- Existence of coupled time steps with additional necessity of a chronological time sequence, e.g. caused by considering the storage level V_t of a water tank for two adjacent time steps t and $t+1 \in T$:

$$V_{t+1} = V_t + \Delta V_t \qquad \forall\, t \in T \tag{2.44}$$

- Necessity for limited time step lengths $\Delta\tau_t$, e.g. caused by the charging or discharging of a water tank with the requirement of constant flows $\dot{V}_t$ in each time step $t \in T$:

$$\Delta V_t = \int_{\tau_t}^{\tau_t+\Delta\tau_t} \dot{V}(\tau)\, d\tau = \dot{V}_t \cdot \Delta\tau_t \qquad \forall\, t \in T \tag{2.45}$$

3 Optimization-Supported System Design

The aim of this thesis is to contribute to the establishment of quantitative, modern algorithmic methods for the design of technical systems. Apart from just introducing these methods, it is necessary to integrate them into the design process. Therefore, a suitable systematic design approach is discussed in the following. Subsequently, an overview of related work is given.

3.1 Systematic Design Approach

In an attempt to automatically find optimal pump system designs, Pelz et al. (2012) proposed a systematic design process in order to combine planning and engineering approaches with algorithmic optimization. Based on the established research field of Operations Research (OR), the term Technical Operations Research (TOR) is used for the methodology.

The traditional system design workflow can be roughly divided into two separate stages: finding a system topology using either experience or some kind of heuristic first and afterwards determining an initial control strategy for the given topology that can be further improved using simulation tools (Altherr et al., 2016a). In contrast to this, one major principle of TOR is not to subsequently optimize the operation of one or several system proposals, i.e. with respect to a given system topology that is unquestioned at this point in time, but to find an optimal system proposal by simultaneously evaluating a large set of implicitly described systems and their operation without being dependent on an a priori defined system topology.

By guiding the designer through specific steps, the approach prepares the generation and solution of an optimization program and structures the application of the optimization results to reality. To enable this, the system design process is divided into seven consecutive steps, which are classified into two phases: a deciding phase and an acting phase. To visualize the approach, the so-called TOR pyramid, as shown in Figure 3.1, is often used. As illustrated, the degree of detail is continuously refined from step to step. These steps are meant to supplement traditional approaches in order to streamline the planning process, facilitate the communication between the interest groups involved and hence catalyze the generation of optimal solutions. Therefore, the TOR methodology does not intend to replace other established design processes, such as VDI 2221, but rather to complement them (Dörig et al., 2014).

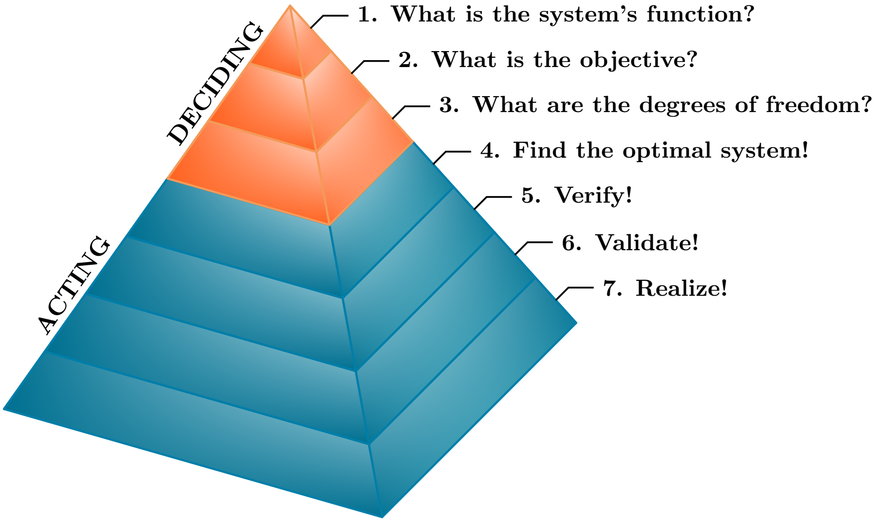

DECIDING
ACTING
1. What is the system's function?
2. What is the objective?
3. What are the degrees of freedom?
4. Find the optimal system!
5. Verify!
6. Validate!
7. Realize!

Figure 3.1: System design with TOR visualized by the TOR pyramid (based on Pelz et al., 2012)

In the following, the individual steps of the TOR approach are discussed in detail. This thesis especially examines the first four steps of this approach. However, the remaining steps of verification, validation, and realization are also of high importance and are therefore also explained in the following for the sake of completeness.

1. What is the system's function? In this step, the basis for the following steps is formed. The basic function of the system is defined and the requirements for the system are precisely determined by providing expected load profiles. Typical basic functions in this context are for example the distribution of fluid or heating and cooling. The formulation of the requirements is crucial because only if a system is able to maintain its function with regard to the occurring loads, it fulfills its intended purpose.

2. What is the objective? The assessment of a system is only possible with regard to the intended objective. This step is of great importance since the objective massively influences the final solution of the problem. The objective can vary depending on the interest groups involved. For example, the objective of an investor can be a low net expense. The operator, in contrast, could consider a high availability, while a state institution could focus on a low energy consumption or low pollutant emissions as a priority. Since these objectives can be conflicting, a completely different system may emerge depending on the decision made in this step, but each of these systems would be optimal with regard to the respective objective. For this reason, the definition of the objective can also be seen as a subjective influence on the optimal system and therefore has to be formulated in agreement with all relevant interest groups. However, this step is often neglected in practice.

3. What are the degrees of freedom? The last step of the deciding phase is to determine the degrees of freedom. This defines the limits within which a system can be optimally designed but also offers room for possible variations. In other words, the "playing field" for the optimization has to be staked out by the designer. In the case of a technical system, the playing field consists of a preselected set of components that may be used for the system, their characteristics, rules for combining these components to a system, and the general prevailing technical and physical relationships for the system as a whole. Afterwards, an algorithm takes over the task of selecting components from the defined set of different components and making optimal use of them for the overall system. Hence, the delimitation of the playing field represents an important restriction for the possible solutions and must be carefully defined in mutual consultation with all interest groups. It must be clear that the approach will only find technical solutions that are part of the playing field and therefore will not replace human imagination or find creative solutions beyond the possible solutions. This underlines the intention of this approach, which has to be seen as a decision tool and should not replace the decision maker. At the end of the deciding phase, all decisions by the system designer that influence the optimal system have been made. Thus, the description of the desired system and the associated optimization task is completed.

4. Find the optimal system! After completing the decision phase, the next step is the computation of a system proposal. This involves decisions regarding the topology of the system as well as the control parameters. Making these decisions and determining a system proposal is done by setting up optimization models, preferably as MILPs, and applying algorithms to solve them (Ederer et al., 2014). The application of MILPs has two major advantages: If there is a solution for the instance under consideration, it can be found in finite time, and moreover, verifiable globally optimal solutions can be obtained (Dörig et al., 2014). However, a global optimum with respect to the initial decisions made in the deciding phase cannot always be found within reasonable time. Nevertheless, it is not necessarily a question of finding the optimal system but of generating the best possible system considering the time aspect and especially being able to estimate its quality. In practice, systems that are proven to be among the best percentage of possible solutions are often more than sufficient.

5. Verify! Following the algorithmic search for a system, the system proposal needs to be verified by the system designer in order to ensure it is technically and physically correct. This step stands in close interaction with the previous step. The verification is carried out by setting up "physical-technical-economic models with concentrated parameters", so-called zero-dimensional models, e.g. using the Modelica[1] modeling language (Ederer et al., 2014).

6. Validate! In this step the system proposal is validated. It must be ensured that the chosen model and its level of detail adequately reflect reality and that the system proposal can be implemented in practice. For this purpose, different approaches can be applied depending on the application. It is possible to use three-dimensional computational methods, such as the Finite Element Method (FEM), carry out experiments or conduct field studies (Ederer et al., 2014).

7. Realize! After verification and validation, the final step is the realization of the system proposal. For this, the systematic execution of all previous steps ensures that all basic prerequisites for the optimization have been covered and that a structured application of the optimization results to reality is guaranteed.

[1] `https://modelica.org` (accessed May 07, 2021)

3.2 Related Technical Work

The starting point for this section is research published in the context of the TOR approach presented in the previous section. All these works have in common that they refer or otherwise relate to this approach and address partial aspects of the overall design process. Subsequently, further literature is discussed that is thematically related to the topics covered in this thesis, with a focus on the application of mathematical programming techniques in the respective domain. To this end, an attempt is made to highlight the most important characteristics of the individual research areas and to describe exemplary publications that can be used to identify the basic principles and commonly applied concepts.

3.2.1 Technical Operations Research

The publications that have appeared in the context of the TOR approach can generally be classified with respect to two focal points: the technical applications considered and the focus on specific aspects of the general approach. This section follows the former of these classification schemes, which focuses on the different application areas. Accordingly, important aspects with respect to the TOR approach are highlighted in the context of the respective application. Recurring aspects across all application areas are the generation of suitable models of technical systems, their validation, the development of tailored solution methods, the consideration of additional characteristics such as uncertainty, and the superordinate attempt to make the methodology more accessible to engineers and system designers.

Optimization of Water-Based Applications

As mentioned above, Pelz et al. (2012) demonstrated the TOR approach using the application example of a domestic pump system for pressure boosting. In this spirit, the most widely investigated field of application is the optimization of booster stations. For this application, the focus is mainly on appropriate modeling of these systems in combination with the application of standard solvers. Intuitively, the associated optimization problem can be modeled as a MINLP due to the arising nonlinearities and integer decision variables induced by the components involved. However, in order to obtain a MILP and therefore be able to use powerful linear programming algorithms, the nonlinearities can be replaced by linearizations (see e.g. Betz, 2017; Pelz et al., 2012). Nevertheless, the MINLP formulation has also been investigated as an alternative, accepting the computational downsides of the nonlinear formulation in favor of a more accurate representation (see e.g. Pöttgen and Pelz, 2016). In this context, Rausch et al. (2016) studied the performance of different solvers for MILPs and MINLPs with regard to different model formulations for a system with spatially distributed pumps.

The application example of decentralized booster stations for high-rise buildings using a MINLP formulation was further studied in Leise et al. (2018) and Leise and Altherr (2019). In this regard, there is also a recently published comprehensive study by Müller et al. (2020), who investigated three important aspects. Firstly, with regard to the modeling of decentralized booster stations, the authors proposed and compared three different variants: a nonlinear, a piecewise linear approximated, and a piecewise linear relaxed formulation. Secondly, the authors developed a problem-specific algorithm for obtaining globally optimal solutions based on the interplay of the different model formulations. Thirdly, based on the preliminary work in Müller et al. (2019c), the obtained solutions were validated on a test rig that represents a scaled version of the considered system. The validation results showed that the obtained solutions were sufficiently accurate with minor modifications to the model and confirmed the expected savings potential. These findings are also in accordance with previous results reported by Altherr et al. (2016a), who examined an experimental setup using a booster station in combination with an accumulator.

Besides the optimal design, the optimal operation in isolation has also been examined as a partial aspect (see e.g. Groß et al., 2017). This was used, inter alia, to determine the maximal attainable "Energy Efficiency Index" (EEI) for booster stations (Pöttgen and Pelz, 2016) and for the closed-loop control of a tank level control system (Altherr et al., 2016b). Furthermore, Pöttgen et al. (2015) proposed a MILP formulation for finding cost-efficient activation strategies for pumps in existing booster stations that are also robust with regard to the operation under uncertain loads.

In addition to examining uncertainty only in the context of the operation of booster stations, this consideration has also been applied to the system design. Altherr et al. (2015) designed optimal pump systems under consideration of downtime costs due to spontaneous failures by including multiple availability scenarios. Sun et al. (2018) considered optimal booster station design and operation under uncertain load and introduced a penalty for water shortage as an additional cost component. Furthermore, Altherr et al. (2018a) investigated decentralized booster stations for high-rise buildings with the aim of finding cost-efficient K-resilient systems, i.e. the operation can be ensured if at most K components break down. In subsequent work, Altherr et al. (2019) further proposed a solution approach based on the Branch and-Bound method to exploit the special tree-structure of the considered systems.

While the above works focus on models and methods to design optimal pumping systems, Saul et al. (2016) developed a domain-specific, textual language that can automatically generate optimization models, in order to make algorithmic optimization methods accessible for a broader audience. According to the TOR approach, this language includes the possibility to define a selection of components as well as their specifications and properties, the topological degrees of freedom, and different load profiles. Based on this description, a specially developed compiler automatically generates the associated optimization model, which is then solved by external standard solvers. Subsequently, a simulation model is created that can be used for validating the obtained solutions using common simulation tools. Another noteworthy feature is that the compiler can analyze the system model and provide human-readable feedback, e.g. regarding the feasibility of the defined specific requirements for the system.

A field of application adjacent to booster stations is the application example of urban water distribution systems. In this regard, Rausch (2018) used algorithmic optimization for the supply planning of informal settlements within a city, so-called slums, in order to design water supply networks with a combination of different supply options. A special feature is that in addition to pipelines, distribution with vehicles was also considered as an alternative supply option. Besides this, Lorenz and Pelz (2020) proposed a MILP for increasing the resilience of a water distribution system with regard to its topology by using a graph-theoretical resilience index considering both redundancy and robustness.

Compared to applications where the distribution of water is considered in isolation, less research has been done on applications involving water-based heating and cooling systems. In Ederer et al. (2014) and Pöttgen et al. (2016), the operation of the generation side of an existing heating circuit with heat storage tanks was examined and based on this, the authors proposed an alternative system concept that was compared to the existing system. However, the system proposal was not the proven globally optimal system but was designed on the basis of engineering experience. Furthermore, Meck et al. (2020) investigated and optimized an existing industrial cooling system. However, the consideration was limited to the optimization of fluid distribution, i.e. pump placement and operation. The authors did not explicitly address thermal consideration in the optimization.

Optimization of Other Fluid-Based Applications

Besides water-based pumping systems, other applications in the field of fluid systems have been explored. A model for the optimization of the topology of a hydrostatic transmission system was presented in Dörig et al. (2014). Based on this, Altherr et al. (2017a) extended the examination of hydrostatic transmission systems to a multi-criterial setting in which the wear reduction of system components was taken into account as an conflicting objective to a low initial investment. In subsequent work, this was revisited by Altherr (2016), who developed a heuristic solution approach based on a combination of Simulated Annealing and Dynamic Programming (DP) to generate good, feasible solutions in reasonable time because the instances under consideration were beyond the capabilities of standard solvers. Furthermore, Altherr (2016) used dual bounds obtained by Lagrangian Decomposition to assess the primal solutions.

Another application example in the domain of fluid systems is paper recycling. In this context, Fügenschuh et al. (2014), who examined the optimal layout for the application example of sticky separation in waste paper processing, built the foundation for subsequent work. In their paper, a nonlinear formulation for the simultaneous selection of the network topology as well as the optimal settings of the considered technical components involved, i.e. the so-called separators, was proposed for the steady state. The authors further derived different piecewise linearizations in order to handle the arising nonlinear relationships. Based on this work, Müller et al. (2019a) extended the model, inter alia, by the consideration of the system's energy consumption. Furthermore, in Müller et al. (2019b), a graphical user interface (GUI) to provide an entry point for designers and operators without more profound knowledge regarding the applied optimization techniques was presented. The GUI was especially designed to guide the user through all necessary steps and present the resulting solutions in an intuitive way. To achieve this, the authors proposed a suitable model for the intended application that is adaptable with respect to certain individual constraints.

Moreover, the examination of the optimal design for the example of ventilation systems for buildings was covered by Altherr et al. (2017b) and Schänzle et al. (2015). Instead of preselecting a discrete set of components, the authors used affinity and scaling laws to describe the technical components. Based on this work, Leise et al. (2019a) extended the consideration by incorporating resilience, i.e. K-resilience, as an additional feature.

Optimization of Non-Fluid-Based Applications

Other fields of application that have been investigated besides fluid-based systems are the optimal design of mechanical transmission systems and the optimization of lattice structures. The optimal design of gearboxes, or more precisely dual-clutch transmission systems, from scratch using MINLPs was examined by Altherr et al. (2018b) and Dörig et al. (2016). Based on this, Leise et al. (2019b) combined a MINLP formulation with an unsupervised machine learning algorithm to generate optimal multi-gear transmission system designs for battery electric vehicles that are robust against load uncertainties. Reintjes and Lorenz (2020) examined the optimization of lattice structures in the context of additive manufacturing. They proposed two different MILPs, one focusing on powder-based additive manufacturing and one focusing on support-free lattice structures for additive manufacturing processes that normally require support structures. In addition, a domain-specific approach to algorithm-based design was presented to bridge the gap between the scientific disciplines of mathematics and engineering in this research area. Furthermore, resilience as an additional feature for load-carrying structures was examined in Altherr et al. (2018a).

3.2.2 Optimization of Water Distribution Systems

In this section, a short overview of the related research area of water distribution network optimization is given. The consideration mainly focuses on the application of mathematical programming techniques. For further insight, D'Ambrosio et al. (2015) offer a comprehensive review in this regard, with a special focus on drinking water supply networks. Besides discussing only applications in the context of mathematical programming and in order to adequately address other aspects covered in this thesis, i.e. metaheuristics and resilience considerations, also separate brief overviews for both are given at the end of this section, even though the considered works are (partly) outside the domain of mathematical programming.

According to D'Ambrosio et al. (2015), literature regarding the optimization of water distribution systems can be divided with respect to two different tasks: optimal operation and optimal design. The optimal operation problem aims at operating the system components, e.g. pumps and valves, of a given fixed system over a certain time horizon such that the customer demands are met while the operational costs, typically related to the components' power consumption, are minimized. For the optimal design problem, there is a distinction between the terms design and layout problem in related literature. As pointed out by De Corte and Sörensen (2013), the layout problem deals with a more strategic decision level, involving decisions with respect to the topology, e.g. the system's connectivity and/or the placement of components, while the design problem assumes a fixed topology of the network and the consideration only involves decisions on a more tactical level. In general, this includes decisions with respect to the selection of pipe diameters and/or pipe material as well as the selection of types and/or sizes for active components, such as pumps and valves (Altherr et al., 2019). However, the design problem often only involves the consideration of pipes since many investigations focus on so-called gravity-fed systems, where active components are not considered (De Corte and Sörensen, 2013). Beyond that, the combined optimization of the operation and the design and/or layout from scratch has not yet been extensively explored (Altherr et al., 2019). Also, the distinction between the terms "design" and "layout" only applies to this section and the term "design" in this thesis otherwise covers aspects of both.

Optimization of Operation

In the context of optimal operation, there are different problems considered in literature. These are the optimal pump operation, optimal system operation for water quality purposes and valve control in combination with the former two (Mala-Jetmarova et al., 2017). Here, the focus is on literature focusing on optimal pump operation, also called optimal pump scheduling, as well as combinations of optimal pump scheduling and valve control. For a more detailed overview with regard to the optimal operation also including works that do not apply mathematical programming techniques, see Mala-Jetmarova et al. (2017).

Optimal pump scheduling can be modeled as a (non-convex) MINLP due to the nonlinear (non-convex) pressure-flow relationship and binary variables that represent the on/off state of components. Since these problems are in general hard to solve and state-of-the art solvers do not scale up well, different simplifications are common in literature: dropping the time dimension, using piecewise linear approximations or ignoring discrete decisions (Bonvin et al., 2017).

The optimal stationary operation, i.e. dropping time-couplings and considering a fixed point in time, was examined by Gleixner et al. (2012). The associated MINLP involved nonlinear relationships, such as pressure losses in pipes due to friction and pump characteristics as well as discrete activation decisions for components. As for active components, constant-speed pumps and continuously adjustable valves for reducing pressure were considered. The water tanks of the network had fixed initial filling levels due to the stationary operation. For validation, the authors studied two real-world examples that could be solved using problem-specific presolving and a standard nonlinear solver.

The optimization of the dynamic operation using a MILP formulation based on piecewise linear approximations of nonlinear relationships was tackled by Morsi et al. (2012) and Geißler et al. (2011) in joint work. The model involved discrete decisions for operating the components and the arising nonlinearities were approximated using the incremental method. In contrast to Gleixner et al. (2012), the authors considered constant-speed as well as variable-speed pumps and different valve types. Furthermore, additional features were incorporated. These included terminal filling levels of water tanks and the necessity for the tanks to be filled and emptied a certain number of times during the considered time horizon to avoid stagnation, as well as minimum runtimes and downtimes for pumps. Besides the incremental method used by Morsi et al. (2012) and Geißler et al. (2011), later work also adapted different approximation methods. For example, Verleye and Aghezzaf (2013) applied a logarithmic convex combination method presented in Vielma and Nemhauser (2011).

The third simplification, i.e. avoiding discrete decisions, was used by Burgschweiger et al. (2009). Derived from a MINLP formulation, the authors obtained a NLP by using special techniques to incorporate the binary decisions. This was achieved based on the aggregation of multiple pumps in a pump station together with a combined efficiency model as well as the introduction of up- and downtime constraints. For further details in this regard, see also Burgschweiger et al. (2005). Given the solution of the NLP with aggregated pumps, the operation scheduling for the individual pumps could then be determined in postprocessing by using the MINLP formulation. With this, it was possible to compute solutions for large real-world networks over a time horizon of 24 hours in reasonable time using standard solvers. However, it was noted by the authors that the obtained solutions were only locally optimal due to non-convexities still present in the NLP formulation.

Besides the above approaches that attempt to tackle the general problem, Bonvin et al. (2017) focused on a specific class of networks in order to obtain a tractable MINLP formulation for pump scheduling. The considered class of networks included the assumptions of a branched layout (no loops or bi-directional pipes) as well as an aggregated pump station at the source and elevated tanks with flow-control valves at the sinks. By focusing only on this special class of networks, the authors were able to solve a real-world industrial example with a time horizon of 24 hours using a convex MINLP formulation and a standard solver by showing that non-convexities can be relaxed for the considered network type. However, while the considered network class is widespread, other networks were explicitly excluded.

Optimization of Design and Layout

As mentioned above, the optimal design mostly involves the sizing of pipe diameters for a given layout, while other components such as pumps or valves are not considered. In this context, Bragalli et al. (2012) used a MINLP approach to select pipe diameters from a set of commercially-available options. With this, the authors were able to find good solutions for large instances originating from both literature and real-world examples. However, these are not necessary globally optimal solutions since the solver framework used acts as an heuristic solver for the considered non-convex MINLPs (D'Ambrosio et al., 2015). Other examples include the application of linearizations resulting in a MILP (see e.g. Artina and Walker, 1983, as cited and implemented in Bragalli et al., 2012) or the application of a NLP approach (see e.g. Eiger et al., 1994). A comprehensive review in this regard is given by Mala-Jetmarova et al. (2018), who also discuss optimization approaches other than mathematical programming.

In the context of optimal layout, Pecci et al. (2019) examined the simultaneous optimization of the placement and the operational settings of control valves with the aim of minimizing the average zone pressure. For this, the authors used a MINLP formulation and obtained ϵ-sub-optimal solutions, i.e. solution values that are within $\epsilon \geq 0$ of the optimal value. This was based on using a Branch-and-Bound implementation in order to obtain reliable bounds. The applied approach used linear relaxations for the hydraulic constraints, which were further tightened by a special domain reduction procedure. The approach was tested on two benchmark instances and one real-world example, for which solutions with bounds in the range of parameter uncertainties could be obtained. Instead of investigating the placement of components for an existing network, Lejano (2006) examined the optimal pipeline layout and design of a branched distribution system. The locations of potential customers and their demands as well as the location of a single, aggregated pumping station were used as inputs. Based on the maximization of the net benefit, i.e. the difference between the revenue of sold water and the costs incurred, it was possible to decide whether a customer should be connected to the network or not. For this, the author used a MILP formulation obtained by using linear approximations and further simplifications regarding the generally nonlinear relationships of the considered investment and operational costs.

Metaheuristics

Besides global optimization methods, heuristics, especially metaheuristics, have been applied in literature since the early 1990s (De Corte and Sörensen, 2013). This development affects all considered areas: optimal operation (see e.g. Hashemi et al., 2014; Lingireddy and Wood, 1998), optimal design (see e.g. Maier et al., 2003; Savić and Walters, 1997), optimal layout with regard to optimal valve placement (see e.g. Paola et al., 2017; Reis et al., 1997), and the joint design and operation (see e.g. Dandy et al., 1994; Ostfeld and Tubaltzev, 2008). Also a wide range of different metaheuristics have been applied, this comprises (but is not limited to) Genetic Algorithms (see e.g. Marchi et al., 2014; Murphy and Simpson, 1992; Savić and Walters, 1997) as the first and most frequently used metaheuristic in this context but also other metaheuristics including Simulated Annealing (see e.g. Costa et al., 2000; Cunha and Sousa, 1999), Tabu Search (see e.g. Cunha and Ribeiro, 2004; Sung et al., 2017), and Ant Colony Optimization (see e.g. Maier et al., 2003; Zecchin et al., 2007). For a comprehensive overview in the context of design problems for gravity-fed water distribution networks, see De Corte and Sörensen (2013).

However, many approaches relying on metaheuristics do not use an explicit mathematical programming formulation (D'Ambrosio et al., 2015). Instead external solvers such as EPANET[2] are often applied to check for hydraulic feasibility (De Corte and Sörensen, 2013). For these, there is a drawback with regard to large networks because of the rapidly increasing number of necessary objective function evaluations by the hydraulic solver (Pecci et al., 2019).

An example for the combination of metaheuristics and mathematical programming was examined in Cai et al. (2001). The authors used a Genetic Algorithm to fix a set of complicating variables in their nonlinear optimization model for water management, resulting in a linear formulation that could be solved using standard techniques. With this, the authors were able to find favorable solutions for large water management instances in reasonable time. However, even though heuristic

[2] `https://www.epa.gov/water-research/epanet` (accessed May 07, 2021)

approaches may oftentimes yield good solutions, optimality cannot be guaranteed in general, not even locally (Pecci et al., 2019).

Resilience

Recently, there have also been increased efforts to include resilience considerations for the design of water distributions systems. According to Mala-Jetmarova et al. (2018), resilience in this context can be defined as the ability to "adapt to or recover from a significant disturbance", where the disturbance can be caused by either an internal, e.g. component failure, or an external reason, e.g. a natural disaster. However, there is as yet no specific, universally agreed definition with regard to water distribution system due to the novelty of the research topic (Mala-Jetmarova et al., 2018).

Due to the lack of a universal definition, there are many different resilience measures as well as many different approaches for assessing resilience. However, mostly simulative approaches are used in this regard (Altherr et al., 2019). Examples include Herrera et al. (2016), who examined resilience with respect to a topological perspective based on an implementation of the K-shortest paths algorithm, and Meng et al. (2018), who proposed an analysis framework for assessing the correlations between resilience and topological features using the application example of water distribution systems.

In the context of mathematical programming, Marinho et al. (2020) presented a MILP for decision support with regard to the resilient design of water distribution systems that are affected by possible disturbances. For this, the authors focused on three resilience capabilities: absorption, adaptation and recovery. The goal was to minimize the investment with respect to different options of actions in order to improve the system's resilience capabilities. These actions could be performed either preemptively or retroactively after a disruption. However, the model provided many simplifications. For example, the authors did not consider any hydraulic constraints, i.e. the consideration of pressure was neglected.

Publications that, in contrast, also include hydraulic properties are the already mentioned works of Altherr et al. (2018a), Altherr et al. (2019) and Lorenz and Pelz (2020). Since these have already been discussed in the previous section, they are not discussed again here but do also apply in this context. Another study explicitly focusing on mathematical programming while considering hydraulic relationships was published by Ulusoy et al. (2020). The authors considered the problem of adding new pipes from an a priori defined set of possible pipes to a given layout in order to enhance resilience with regard to failure events while trying to minimize the disadvantageous consequences arising in the context of so-called leakage management during the failure-free operation. In this regard, a MINLP for the arising problem of simultaneously optimizing the addition of favorable pipes for redundancy and the control of existing pressure valves was presented, aimed at minimizing the average zone pressure that is used as a surrogate for a favorable leakage management. The authors solved instances based on examples from literature and sections of real-world networks using an adapted spatial Branch-and-Bound procedure. The hydraulic resilience of the system was then evaluated in postprocessing by performing a critical link analysis focusing on reserve capacity as an appropriate resilience measure.

3.2.3 Optimization of Gas Networks

Another research area adjacent to the optimization of water distribution systems is the optimization of gas networks. A general overview on gas network optimization is given by Ríos-Mercado and Borraz-Sánchez (2015). While there are manifold related optimization problems, an interesting class of problems arose due to the deregulation of the gas industry. As a consequence of this deregulation, network operators and gas vendors, which before had been a joint company, have to be different companies now. In this context, new types of problems dealing with the identification and allocation of free network capacities emerged in literature (Ríos-Mercado and Borraz-Sánchez, 2015). According to Fügenschuh et al. (2013), who discuss the arising problems, these include the validation of so-called nominations, the verification of so-called booked capacities, the determination of available freely allocable capacities, and topology planning for extending networks.

However, since the simplification of incompressible fluids, which is used in this thesis, does not apply to the considered gases, the approaches used in gas network optimization are only to some extend applicable to this thesis and vice versa. Nevertheless, there are noteworthy approaches in literature especially with regard to the nomination validation problem. The nomination validation problem deals with the optimization task of finding optimal settings for active components of a given gas network with regard to a nomination, i.e. the amount of gas the consumers announce to feed into or extract from the network, such that all physical, technical and legal constraints are met, with the objective typically being to minimize the costs of transportation (Pfetsch et al., 2015).

Particularly relevant with respect to this thesis are works that consider the piecewise linearization of the arising nonlinear relationships in order to obtain MILP formulations that can be tackled by linear programming techniques. In this context, Martin et al. (2006) examined the stationary problem. Using piecewise linear approximations based on a generalization of the concept of Special Ordered Set (SOS) constraints, the authors were able to solve large MILPs. However, the authors also pointed out that this was only a first step, since although large parts of the network exhibit a stationary behavior other parts are dynamic. In subsequent work, Mahlke et al. (2010) therefore investigated the transient case and focused on the time-dependent aspects of the problem. Again, the concept of SOS constraints was used to approximate nonlinearities and the problem was solved using Branch-and-Cut. For the Branch-and-Cut algorithm, a standard Simulated Annealing implementation was used as a primal heuristic. Interestingly, the authors reported that feasible solutions for the considered test instances with runtimes limited to 15 minutes were only obtained when Simulated Annealing was used. This highlights the benefits of providing good primal solutions.

Furthermore, the study by Geißler et al. (2011), which was already discussed above in the context of water distribution systems since it provides examples for both applications, can be mentioned. For gas networks, the same approach as for water distribution systems, i.e. the approximation of nonlinear constraints by the incremental method, was used. Besides this, the paper dealt with the details of modeling the considered problem with regard to gas network optimization, which is, however, beyond the scope of this thesis. In this context, the associated contribution in Geißler et al. (2012) focused more in-depth on the presentation and performance comparison of different linearization techniques for gas (and water supply) network optimization. In addition to this comparison, the authors proposed methods to a priori estimate the linearization errors and refine the linearization until a certain predefined error bound is reached. Furthermore, it was shown how to obtain piecewise linear relaxations instead of approximations.

3.2.4 Optimization of Energy Systems

Another research field related to this thesis is the optimization-based synthesis of energy systems, typically operating as cogeneration systems for the simultaneous production of heat and power or trigeneration systems with coupled cooling (Andiappan, 2017). Similar to water network optimization, different levels are considered. According to Frangopoulos et al. (2002), these are the synthesis level, the design level, and the operational level. At the synthesis level, component selection as well as the layout of the system is determined. At the subsequent design level, the focus is on decisions related to the sizing of components or operating limits. The third and last level, the operational level, comprises operation related decisions. While there are hierarchical approaches, considering a certain level in isolation, integrated approaches, which are considered here, deal with the optimization of all three levels simultaneously in order to obtain synergies between the individual levels (Andiappan, 2017). For more insights into optimization approaches for energy systems, see the review by Andiappan (2017). However, while contributions in this field also focus on heating and cooling (combined with power generation), the approaches intentionally consider a higher level of aggregation for the synthesis task than it is the scope of this thesis.

In the context of energy systems, for example, Ashouri et al. (2013) proposed a MILP-based design framework for the selection and sizing of a smart building energy system, while also considering the operation of the system. For the framework, the authors considered a variety of different technologies such as thermal and electrical storage, heating and cooling components as well as renewable energy sources. The approach was successfully tested on the application example of a commercial building using standard MILP solvers. However, it was assumed that the considered technologies were continuously sizable instead of using discrete component sets. Hence, only one aggregated component of a given technology, representing the set of individual components, was considered. Therefore, this approach can potentially result in large inaccuracies regarding the actual investment as well as the operation of individual components (cf. Schütz et al., 2016). Furthermore, the considerations were based on the assumption that the different technologies only consume, provide or store energy with regard to four different energy streams. Hence, the representation is limited to the consideration of energy balances. Beyond this, explicit quality levels of the water-based flows, e.g. temperature or pressure levels, were not considered.

Voll et al. (2013) proposed a framework for the automated superstructure-based synthesis of distributed energy supply systems with a particular focus on the superstructure generation. The term “superstructure” refers to a representation that contains all possible solution alternatives, i.e. all considered components and possible connections. Therefore, the superstructure-based synthesis problem comprises three subsequent steps: definition of a superstructure, mathematical modeling based on the superstructure, and solving the arising optimization problem (Voll, 2013). The optimization model was automatically generated from the superstructure representation using a generic, component-based model formulation and model-templates. The model itself was based on a model formulation proposed by Yokoyama et al. (2002) for the simultaneous optimization of topology, sizing and design while assuming a given piping layout for the distribution. In contrast to Ashouri et al. (2013), variable part-load performance for the components was incorporated in the model. To obtain a MILP, the nonlinear cost and component performance relationships were approximated by piecewise linear functions. While the formulation in Voll et al. (2013) dealt with conventional component technologies, it was later extended by Bahl (2018) to include the consideration of thermal storage and renewable energy sources. For the solution, Voll et al. (2013) chose an iterative approach since it was not certain a priori how many different components of one type should be part of the considered superstructure. Hence, the authors started with a superstructure containing each suitable component type once. This was then repeated iteratively by adding further components of the same type until no further improvement of the objective value could be observed. For both the so-called grassroots synthesis, i.e. the system design from scratch, and the retrofit synthesis, i.e. the extension of an existing system, the authors successfully tested their approach on a real example from the pharmaceutical industry. However, the focus was rather on a conceptual level. Hence, only standardized components with continuously adjustable size were assumed. Furthermore, the part-load performance of all components of one type was assumed to be the same, independent of the specific component or its size. Besides that, again only energy balances were considered and temperature and pressure levels of the water-based flows were assumed to be constant and therefore neglected. Because of this, also corresponding aspects such as mixing temperatures and temperature-dependent component performance were neglected. Hence, the consideration of the interconnection of components, i.e. parallel and serial connections, could be neglected since it had no influence on the system behavior for the assumptions made (see the discussion in Voll, 2013). Besides the superstructure-based optimization approach, Voll et al. (2012) proposed a superstructure-free approach for the same setting. For this, design alternatives were generated using an evolutionary algorithm.

3.2.5 Model Predictive Control of HVAC Systems

Finally, there is the research area of Model Predictive Control (MPC). According to Risbeck (2018), MPC is a form of control in which a system model is combined with forecasts of external parameters and the resulting optimization problem of finding control decisions for a certain (moving) time horizon is solved online and in real-time. The optimization is carried out at each considered time step, with the initial state always being the current, measured state of the system. Hence, only the first control decision, i.e. the one until the next considered time step, of the obtained control sequence for the whole time horizon is applied to the system before the procedure is repeated at the next time step. In the following, a combination of two special cases for MPC problems is assumed. Firstly, MPC problems with a cost function, so-called economic MPC, and secondly, MPC problems with both continuous and discrete actuators are considered (see Risbeck, 2018).

While MPC can be used for different applications, the scope considered here is on the application to Heating, Ventilation and Air Conditioning (HVAC) systems. A general overview of MPC applied to HVAC systems is given in Afram and Janabi-Sharifi (2014). Following the definition of Rawlings et al. (2018), large-scale HVAC systems consist of two parts, the airside and the waterside subsystem. The airside subsystem comprises the different building zones with air handling units used for the temperature control of the building. The waterside subsystem typically comprises a central plant used to generate the necessary heating and cooling. The overall control objective is to find temperature setpoints for the different building zones of the airside subsystem as well as operation schedules for the equipment of the waterside subsystem. However, often both are considered separately in literature. As it is more relevant for the context of this thesis, the focus is on the waterside subsystem. Nevertheless, it is important to mention that works in the context of MPC aim to provide an optimal control focusing on the online and real-time aspect, whereas the emphasis in this thesis is on the design aspect and the integration of estimated load data is rather used in order to evaluate favorable system designs for the intended use.

In Deng et al. (2013) and Deng et al. (2015), a MPC approach was applied to the on/off operation of a central electric chiller plant with Thermal Energy Storage (TES) in order to meet the cooling demand of a university campus while minimizing energy costs. The authors proposed a MINLP formulation that had to be solved at each considered time step. However, because of the real-time aspect, linearizations were used to obtain a MILP instead, which was solved using a standard solver. For validation purposes, the actual system behavior was substituted by a simulation based on historical data and the approach was compared to a baseline strategy obtained by a simple greedy-search-based heuristic. While Deng et al. (2013) considered a given fixed TES operation profile with respect to on- and off-peak electricity prices, Deng et al. (2015) extended the solution approach by determining the TES operation profile via DP in a preceding step. Similarly, Ma et al. (2009) applied a two-stage approach using Branch-and-Bound to obtain a real-time control for a chiller system with thermal storage. In the first stage, they fixed the tank operation profile, while in the second stage a reformulated NLP was solved using standard solvers. A slightly different setting was used in Kashima and Boyd (2013), who also examined a chiller system with TES to provide cooling for building air conditioning. Instead of the typical system arrangement involving one large TES connected in parallel to the chiller bank, multiple TESs were connected in series to individual chillers, while there were also additional support chillers to cover peak loads. Besides that, the authors did not assume fixed energy prices. Hence, both future cooling demand as well as energy prices were part of the prediction. With regard to the considered optimization model, the authors considered a simple representation in which only energy balances were taken into account. For this reason, the model could be natively stated as a much smaller MILP and could be solved using a customized Branch-and-Bound framework based on LP relaxations.

While the above works assumed the cooling load as a parameter and therefore decoupled the waterside and airside subsystems, Rawlings et al. (2018) considered both subsystems simultaneously. For this, the authors used a hierarchical decomposition approach. For the high-level problem, aggregated models for the air- and waterside were used. The obtained results were then passed down to the more detailed low-level models for the water- and airside subproblems. As mentioned above, airside models are out of the considered scope. However, for a detailed discussion, see Patel et al. (2016a) and Patel et al. (2016b). With regard to the waterside low-level problem in general, heating as well as cooling components were considered. However, due to different component technologies in combination with different energy streams and energy sources, a simplification to the representation was made. The representation was broken down to the scheduling of "generic generators", representing components of the central plant, and the consideration of energy balances for "abstract resources". For an in-depth representation of the waterside model and its characteristics, see also the preliminary works Risbeck et al. (2015) and Risbeck et al. (2017). With regard to the operational behavior, piecewise linear approximations were used to obtain a MILP that could be solved using standard software for a large-scale example.

4 Modeling of Fluid Systems

All fluid systems considered in this thesis have two things in common: First, each system conveys a certain amount of fluid that moves through a network of connected pipes and other components. Second, there is a difference in potential that causes the fluid to move. This potential is the pressure, which is the driving force in fluid systems. The relevant physical quantities to describe such systems are therefore the volume flow rate $\dot{V}$ and the pressure p. In general, the term fluid comprises both liquids and gases. However, the content of this work is limited to incompressible fluids and is explained in particular using water as an example. Hence, incompressibility is assumed, which is a typical simplification. Furthermore, it is common for water-based fluid systems to express pressure in terms of head H, which is an equivalent representation if an incompressible fluid is assumed (Geißler et al., 2011). Accordingly, the conversion results from using $p = H \cdot \rho \cdot g$, with constant density ρ and acceleration due to gravity g.

The general system synthesis task considered here can be stated as follows: Given a set of available technical components as well as a technical specification of load collectives, compare all valid systems and choose the one for which the lifetime costs, i.e. the sum of the investment and the expected energy costs, are minimal. In this context, a system is called a valid system if it is capable of covering the anticipated loads. It is assumed that the transition times and therefore also the transition costs between the load changes are negligible compared to the total costs. Hence, corresponding models can be stated as quasi stationary. Each load out of the load collective is called a load scenario. A load scenario consists of two components: a probability that represents the portion of the system's operational lifetime in which this scenario is assumed to occur and the required values for the respective physical quantities at specific points in the system.

The decision making can be abstracted in two ways. On the one hand, it can be stated using linear (and nonlinear) constraints as a two-stage MILP (or MINLP). In this context, the decisions of the optimization problem can be described by first and second stage variables. In the first stage, it has to be decided whether a component is needed and thus bought. In the second stage, a bought component can be controlled to cover all load scenarios during the system's operation. On the other hand, the problem can be abstracted as a source-target-network (G, S_G, T_G), with a graph $G = (V, E)$, vertices V and edges E, where $S_G, T_G \in V$ are distinguished sets of vertices, namely the sources and the sinks of the network. Here, edges represent the components from the set of available options and vertices represent certain measurement points. The complete graph consisting of all components that may be installed and all specified connections plus the sources and sinks contains every possible system. Therefore, each system can be modeled by a subgraph of the complete graph in order to present the decisions made for the system.

4.1 Components of Fluid Systems

In this work, three different component types for fluid systems are considered. The general description as well as the most relevant physical and technical features are discussed in the following subsections to provide the basis for the representation as an optimization model.

4.1.1 Pumps

In general, pumps have an opposite relationship between their volume flow rate $\dot{V}$ and their developed head (or equivalently pressure increase) ΔH. Theoretically, the maximum head is reached at zero volume flow and the head decreases with increasing volume flow. Additionally, the power consumption P of pumps increases with rising volume flow. In contrast to constant-speed pumps, variable-speed pumps can also be operated at different rotational speeds n. The relationship between a pump's volume flow rate, head, power consumption, and rotational speed is given by the so-called affinity laws:

$$\dot{V} \sim n, \; \Delta H \sim n^2 \text{ and } P \sim n^3 \tag{4.1}$$

Accordingly, the specification of any two of these variables determines the remaining variables (Altherr et al., 2019). The relationship is also manifested in a pump's respective characteristic map, see Figure 4.1. Furthermore, the operation of a pump can be described by quadratic and cubic approximations with regression coefficients α_i and β_i to determine the head and power consumption for a given flow-speed-tuple (see Ulanicki et al., 2008):

$$\Delta H = \alpha_1 \cdot \dot{V}^2 + \alpha_2 \cdot \dot{V} \cdot n + \alpha_3 \cdot n^2 \tag{4.2}$$

$$P = \beta_1 \cdot \dot{V}^3 + \beta_2 \cdot \dot{V}^2 \cdot n + \beta_3 \cdot \dot{V} \cdot n^2 + \beta_4 \cdot n \tag{4.3}$$

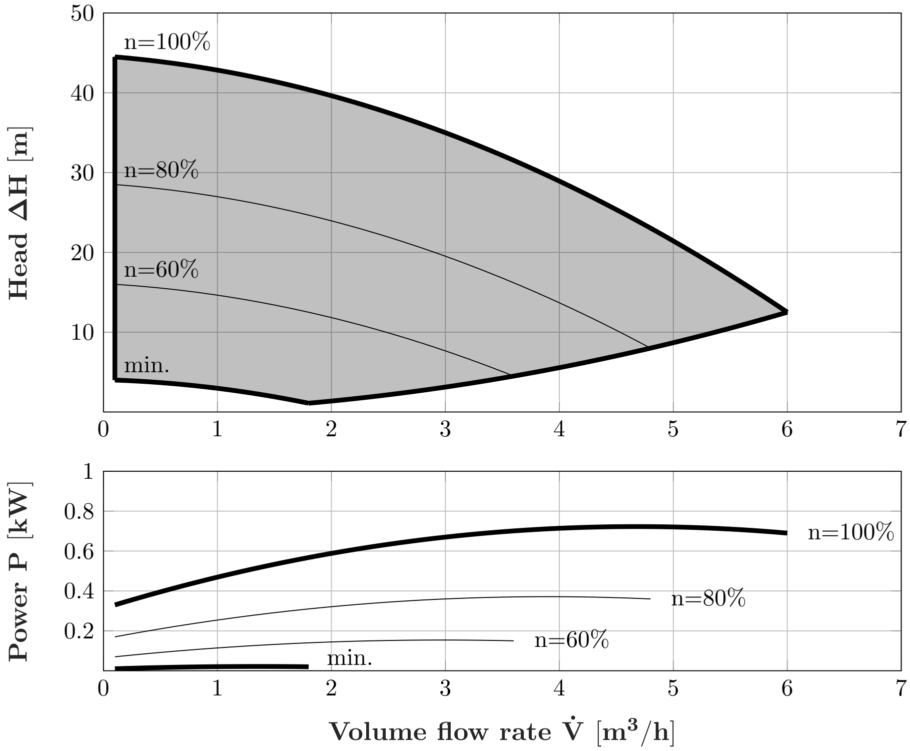


Figure 4.1: Exemplary characteristic map of a variable-speed pump (based on Wilo SE, 2020b)

Single pumps or whole subsystems of pumps can be connected pairwise either in series or in parallel. If, on the one hand, pumps are connected in series, the obtained head results from the sum of the individual heads, while each pump handles the entire volume flow rate:

$$\dot{V}_{series} = \dot{V}_1 = \dot{V}_2 \tag{4.4}$$

$$\Delta H_{series} = \Delta H_1 + \Delta H_2 \tag{4.5}$$

If, on the other hand, pumps are connected in parallel, the total volume flow rate is the sum of the individual volume flows and each pump generates the same head:

$$\dot{V}_{parallel} = \dot{V}_1 + \dot{V}_2 \tag{4.6}$$

$$\Delta H_{parallel} = \Delta H_1 = \Delta H_2 \tag{4.7}$$

4.1.2 Pipes and Gate Valves

Pipes serve to convey flow between spatially separated components. There is a pressure loss due to friction along pipes, which depends quadratically on the volume flow rate through the pipe according to the Darcy-Weisbach equation. If further components, such as fittings or valves, are implicitly included, the respective losses have to be taken into account, see Section 2.1.2.

In contrast to pipes, gate valves can be used to open or close paths in the system. According to Pöttgen et al. (2016), it is a simple component with two states. Either the valve is completely open, in which case the flow can pass freely, or it is completely closed. A closed valve prevents flows between its inlet and outlet and the adjacent pressures become decoupled.

4.2 Optimization Model for Fluid Systems

The optimization model for fluid systems presented here is inspired by models available in literature (see e.g. Betz, 2017; Pelz et al., 2012; Pöttgen et al., 2016). It serves as a starting point for the step-by-step extension according to Figure 1.1. The model is divided into three parts: objective function, general system constraints, and component-specific constraints. The general system constraints deal with the structure and general operational restrictions of the systems as well as their physical behavior. The consideration of the component behavior is dealt with separately. Hence, different components can be treated as placeholders or black boxes and their interactions in the system can be considered independent of their specific internal behavior. This helps to ensure that new component types can be added without having to question the general behavior of the system or its structure. For the sake of a simple and compact representation, nonlinear relationships that have to be linearized are represented by expressions of the form $y = \phi(x_1, .., x_n)$. Furthermore, it is important to mention that the symbols used within the model should not be confused with the symbols used in other contexts, as they may differ in some cases. For the model, for instance, variables are always denoted by lowercase letters and parameters by uppercase letters. Therefore, all sets, variables and parameters used in the model are presented and described in Table 4.1. If applicable, the dimension of an entry is specified using the base quantities mass (M), length (L), currency (C) and time (T). The entries only applying to the component-specific parts are separated by dashed lines.

Table 4.1: Variables, sets and parameters of the optimization model for fluid systems

Symbol	Range	Dimension	Description
$b_{i,j}$	$\{0,1\}$		Purchase indicator of edge $(i,j) \in E$
$a^s_{i,j}$	$\{0,1\}$		Activation indicator of edge $(i,j) \in E$ in scenario $s \in S$
$p^s_{i,j}$	$\mathbb{R}^+_0$	ML^2T^{-3}	Power consumption caused by edge $(i,j) \in E$ in scenario $s \in S$
$\dot{v}^s_{i,j}$	$\mathbb{R}^+_0$	L^3T^{-1}	Volume flow along edge $(i,j) \in E$ in scenario $s \in S$
h^s_k	$\mathbb{R}^+_0$	$ML^{-1}T^{-2}$	Pressure at vertex $k \in V$ in scenario $s \in S$
$\Delta h^s_{i,j}$	$\mathbb{R}$	$ML^{-1}T^{-2}$	Change in pressure caused by edge $(i,j) \in E$ in scenario $s \in S$
$n^s_{i,j}$	$[0,1]$		Relative rotational speed for pump edge $(i,j) \in Pu(E)$ in scenario $s \in S$
S			Set of scenarios
E			Set of edges
V			Set of vertices
$S_G(V)$	$\subseteq V$		Subset of source vertices
$T_G(V)$	$\subseteq V$		Subset of sink vertices
$Pu(E)$	$\subseteq E$		Subset of pump edges
$Pi(E)$	$\subseteq E$		Subset of pipe edges
$GV(E)$	$\subseteq E$		Subset of gate valve edges
$C^{buy}_{i,j}$	$\mathbb{R}^+_0$	C	Purchase costs of edge $(i,j) \in E$
C^{kWh}	$\mathbb{R}^+_0$	$M^{-1}L^{-2}CT^2$	Energy costs per kilowatt hour of electricity
OLT	$\mathbb{R}^+_0$	T	Expected operational lifetime of the system
F^s	$[0,1]$		Portion of time for which scenario $s \in S$ is expected to occur
$\dot{V}^{max}$	$\mathbb{R}^+_0$	L^3T^{-1}	General upper bound on the volume flow
H^{max}	$\mathbb{R}^+_0$	$ML^{-1}T^{-2}$	General upper bound on the pressure
$\dot{V}^{min/max\,s}_{in/out\,k}$	$\mathbb{R}^+_0$	L^3T^{-1}	Lower and upper bound on the volume flow entering and leaving vertex k in scenario s
$H^{min/max\,s}_k$	$\mathbb{R}^+_0$	$ML^{-1}T^{-2}$	Lower and upper bound on the pressure at vertex $k \in V$ in scenario $s \in S$

Continued on next page

Table 4.1: Continued from previous page

Symbol	Range	Dimension	Description
$\dot{V}_{i,j}^{min/max}$	$\mathbb{R}_0^+$	L^3T^{-1}	Lower and upper bound on the volume flow for edge $(i,j) \in E$
$N_{i,j}^{min/max}$	$[0,1]$		Lower and upper bound on the relative rotational speed for pump edge $(i,j) \in Pu(E)$
$P_{i,j}^{max}$	$\mathbb{R}_0^+$	ML^2T^{-3}	Upper bound on the power consumption by pump edge $(i,j) \in Pu(E)$
$\Delta\mathcal{H}_{i,j}(*)$	$\mathbb{R}$	$ML^{-1}T^{-2}$	Representation of the nonlinear relationship for the change in pressure caused by edge $(i,j) \in E$
$\mathcal{P}_{i,j}(*)$	$\mathbb{R}_0^+$	ML^2T^{-3}	Representation of the nonlinear relationship for the power consumption caused by edge $(i,j) \in Pu(E)$

4.2.1 Objective Function

The objective of the optimization model is to minimize the system's lifetime costs, see Objective (4.8). Here, the lifetime costs are assumed to be the sum of the investment for all components present in the system (left-hand term) and the expected energy costs over the system's operational lifetime (right-hand term). The expected energy costs arise from the weighted sum of the power consumption of each individual component over all load scenarios multiplied by the system's operational lifetime and a cost parameter accounting for the costs per unit of the consumed energy. The cost parameter may be dependent on the actual component due to possibly different sources of energy used to power different components. However, in the presented fluid system model, the only considered energy source is electricity and the cost parameter is therefore the same for all components. Furthermore, the objective function can easily be replaced by other objectives. Examples are the exclusive consideration of the investment, the energy costs, or a detailed consideration of the cost flows as shown in Meck et al. (2020), where an annuity-present value factor is used for discounting the energy costs over the considered lifetime of the system.

$$\min \quad \sum_{(i,j)\in E} \Big(C_{i,j}^{buy} \cdot b_{i,j} \Big) + C^{kWh} \cdot OLT \cdot \sum_{s\in S} \Big(F^s \cdot \Big(\sum_{(i,j)\in E} p_{i,j}^s \Big) \Big) \tag{4.8}$$

4.2.2 General System Constraints

Constraints (4.9) model the general system structure as well as general system restrictions. Intentionally no complete interconnection of all components is considered by default, but possible connections must first be explicitly stated by defining the sets of edges E and vertices V. In addition, it is intended that the binary purchase indicators $b_{i,j}$ are fixed for predefined parts of the system.

The structure of the overall system is defined such that only those component edges that are present in the overall system ($b_{i,j} = 1$) can be used in the individual load scenarios, see Constraint (4.9a). The volume flow along a component edge must be reasonable and it vanishes if the edge is not active ($a^s_{i,j} = 0$) in a particular load scenario, see Constraint (4.9b). This is ensured by using the well-known "big-M" formulation, which can be used to express "if-then-else" conditions (Vielma, 2015). Here, the binary activation indicator $a^s_{i,j}$ is used for case differentiation and the upper bound on the system's volume flow $\dot{V}^{max}$ represents a sufficiently large constant. In the following, this method is applied repeatedly using the different bounds of the associated variables as big-Ms. This ensures that big-M is as large as necessary to avoid unintended restrictions for the model but not excessively large, which could lead to numerical instability. Similarly to the volume flow, the pressure must be reasonable at each vertex but does not necessarily vanish with regard to the operational status of components since it is defined for vertices instead of edges, see Constraint (4.9c). According to the explanations given in Section 2.1.1, the volume flow has to be conserved at all vertices with the exception of source and sink vertices, see Constraint (4.9d). Thus, it is required that the sum of all volume flows entering a vertex equals the sum of volume flows leaving that vertex in each load scenario. In general, it is possible that there are exceptions other than sources and sinks, such as storage vertices that are able to store a certain amount of fluid. For these cases, it might be necessary to introduce a variable $\Delta\dot{v}^s_k$ representing the difference in incoming and outgoing volume flows and modify Constraint (4.9d) accordingly. However, for the setting considered here, it is not necessary. Therefore, the clearer representation is preferred instead. If an edge is active, the pressure propagates along this edge, see Constraints (4.9e) and (4.9f). Hence, the pressures of two vertices connected by an active edge have to be coupled such that the pressure at the end vertex equals the pressure at the start vertex plus the change in pressure caused by the active edge, according to the explanations given in Section 2.1.2. However, if the edge is not active, the pressures at the start and end vertex are decoupled. This is ensured by using a variation of the big-M formulation. With this, the constraint can be turned “on” or “off” based on the value of the binary activation indicator. If, on the one hand, the connecting edge is active, both inequalities together yield equality. If, on the other hand, the edge is not active, the inequalities are switched off and thus no longer have a restrictive effect. This is also frequently used in the following for similar contexts, but it is not explicitly mentioned again. Constraints (4.9g) to (4.9i) enable the setting of target ranges for the incoming and outgoing volume flow and the pressure at certain points in the system or can be used to limit their values. By equating the respective lower and upper bound it is also possible to define target values instead of ranges. This is primarily used to specify the supplied or required volume flow and pressure for source and sink vertices. However, the constraints are not limited to this application and can be used to implement other system restrictions regarding minimum and maximum values of the volume flow and pressure at certain points in the system. If they are not explicitly specified for some of the vertices, the general bounds according to Constraints (4.9b) and (4.9c) hold instead.

$$a_{i,j}^{s} \leq b_{i,j} \qquad \forall\, s \in S,\ (i,j) \in E \tag{4.9a}$$

$$\dot{v}_{i,j}^{s} \leq \dot{V}^{max} \cdot a_{i,j}^{s} \qquad \forall\, s \in S,\ (i,j) \in E \tag{4.9b}$$

$$h_{k}^{s} \leq H^{max} \qquad \forall\, s \in S,\ k \in V \tag{4.9c}$$

$$\sum_{(i,k)\in E} \dot{v}_{i,k}^{s} - \sum_{(k,j)\in E} \dot{v}_{k,j}^{s} = 0 \qquad \forall\, s \in S,\ k \in V \setminus (S_G(V) \cup T_G(V)) \tag{4.9d}$$

$$h_{j}^{s} - h_{i}^{s} \leq \Delta h_{i,j}^{s} + H^{max} \cdot (1 - a_{i,j}^{s}) \qquad \forall\, s \in S,\ (i,j) \in E \tag{4.9e}$$

$$h_{j}^{s} - h_{i}^{s} \geq \Delta h_{i,j}^{s} - H^{max} \cdot (1 - a_{i,j}^{s}) \qquad \forall\, s \in S,\ (i,j) \in E \tag{4.9f}$$

$$\dot{V}_{out\,k}^{min\,s} \leq \sum_{(k,j)\in E} \dot{v}_{k,j}^{s} \leq \dot{V}_{out\,k}^{max\,s} \qquad \forall\, s \in S,\ k \in V \tag{4.9g}$$

$$\dot{V}_{in\,k}^{min\,s} \leq \sum_{(i,k)\in E} \dot{v}_{i,k}^{s} \leq \dot{V}_{in\,k}^{max\,s} \qquad \forall\, s \in S,\ k \in V \tag{4.9h}$$

$$H_{k}^{min\,s} \leq h_{k}^{s} \leq H_{k}^{max\,s} \qquad \forall\, s \in S,\ k \in V \tag{4.9i}$$

4.2.3 Component-Specific Constraints

Three different component types are considered, which are described by Constraints (4.10) to (4.12): pumps, pipes, and gate valves. The description is done separately for each component to ensure the extensibility of the model. In order to obtain a MILP, the nonlinear relationships necessary for describing the component characteristics are modeled as piecewise linear approximations using the techniques presented in Section 2.2.3. This is indicated by using the component-specific expressions $\Delta\mathcal{H}_{i,j}(*)$ and $\mathcal{P}_{i,j}(*)$. For this, associated variables have to be a convex combination of the respective grid points if a component is active in a certain load scenario. However, for reasons of a universal and compact presentation, this is not explicitly shown here but the implementation is straightforward.

Pumps

Constraint (4.10a) ensures that possible restrictions on the minimum and maximum volume flow for a specific pump are met. Similarly, the relative rotational speed of a pump is limited by its minimum and maximum values or zero if the pump is not active, see Constraint (4.10b). This formulation deals with the more general case of variable-speed pumps. However, it is also valid for constant-speed pumps by setting the upper and lower bounds of the relative rotational speed of this pump to one. If a pump is active, the pressure increase caused by the pump depends on the volume flow through the pump and its rotational speed in its respective physical domain according to the nonlinear relationship discussed in Section 4.1.1, see Constraint (4.10c) and (4.10d). If a pump is not active, an arbitrary value in its physical domain is selected instead. However, the pressure increase is forced to vanish by Constraint (4.10e). As with the pressure increase, the power consumption of an active pump depends on the volume flow and the rotational speed, taking into account the physical domain of the pump, see Constraints (4.10f) and (4.10g). If a pump is not active in a certain load scenario, the power consumption by this pump is forced to vanish, see Constraint (4.10h). In combination, the presented constraints ensure that pumps are only operating within the boundaries of their respective characteristic map.

$$\dot{V}_{i,j}^{min} \cdot a_{i,j}^s \leq \dot{v}_{i,j}^s \leq \dot{V}_{i,j}^{max} \cdot a_{i,j}^s \qquad \forall s \in S,\ (i,j) \in Pu(E) \tag{4.10a}$$

$$N_{i,j}^{min} \cdot a_{i,j}^s \leq n_{i,j}^s \leq N_{i,j}^{max} \cdot a_{i,j}^s \qquad \forall s \in S,\ (i,j) \in Pu(E) \tag{4.10b}$$

$$\Delta h_{i,j}^s \leq \Delta\mathcal{H}_{i,j}\left(\dot{v}_{i,j}^s, n_{i,j}^s\right) + \Delta H_{i,j}^{max} \cdot (1 - a_{i,j}^s) \qquad \forall s \in S,\ (i,j) \in Pu(E) \tag{4.10c}$$

$$\Delta h_{i,j}^s \geq \Delta\mathcal{H}_{i,j}\left(\dot{v}_{i,j}^s, n_{i,j}^s\right) - \Delta H_{i,j}^{max} \cdot (1 - a_{i,j}^s) \qquad \forall s \in S,\ (i,j) \in Pu(E) \tag{4.10d}$$

$$0 \leq \Delta h_{i,j}^s \leq \Delta H_{i,j}^{max} \cdot a_{i,j}^s \qquad \forall s \in S,\ (i,j) \in Pu(E) \tag{4.10e}$$

$$p_{i,j}^s \leq \mathcal{P}_{i,j}\left(\dot{v}_{i,j}^s, n_{i,j}^s\right) + P_{i,j}^{max} \cdot (1 - a_{i,j}^s) \qquad \forall s \in S,\ (i,j) \in Pu(E) \tag{4.10f}$$

$$p_{i,j}^s \geq \mathcal{P}_{i,j}\left(\dot{v}_{i,j}^s, n_{i,j}^s\right) - P_{i,j}^{max} \cdot (1 - a_{i,j}^s) \qquad \forall s \in S,\ (i,j) \in Pu(E) \tag{4.10g}$$

$$p_{i,j}^s \leq P_{i,j}^{max} \cdot a_{i,j}^s \qquad \forall s \in S,\ (i,j) \in Pu(E) \tag{4.10h}$$

Pipes

A pipe cannot be deactivated if it is present in the system, see Constraint (4.11a). As with pumps, it is possible that there are restrictions on the minimum and maximum volume flow for a specific pipe, see Constraint (4.11b). The pressure decrease caused by the pipe is a function of the volume flow through that pipe with a priori known characteristics, see Constraint (4.11c). For short pipe lengths, the pressure change can be neglected if necessary. Furthermore, a pipe is not a powered component and thus has no power consumption, see Constraint (4.11d).

$$
\begin{aligned}
b_{i,j} &= a_{i,j}^s && \forall s \in S,\ (i,j) \in Pi(E) && (4.11a)\\
\dot{V}_{i,j}^{min} \cdot a_{i,j}^s \leq \dot{v}_{i,j}^s &\leq \dot{V}_{i,j}^{max} \cdot a_{i,j}^s && \forall s \in S,\ (i,j) \in Pi(E) && (4.11b)\\
\Delta h_{i,j}^s &= \Delta \mathcal{H}_{i,j}\left(\dot{v}_{i,j}^s\right) && \forall s \in S,\ (i,j) \in Pi(E) && (4.11c)\\
p_{i,j}^s &= 0 && \forall s \in S,\ (i,j) \in Pi(E) && (4.11d)
\end{aligned}
$$

Gate Valves

A gate valve being present in the system ($b_{i,j} = 1$) can be closed ($a_{i,j}^s = 0$) during operation to prevent flows and to decouple the pressures at its inlet and outlet. Possible restrictions on the minimum and maximum volume flow for a specific component can be enforced by Constraint (4.12a). In contrast to pipes, the pressure change along this component is neglected, see Constraint (4.12b). It is also not considered to be a powered component and therefore has a power consumption of zero, see Constraint (4.12c).

$$
\begin{aligned}
\dot{V}_{i,j}^{min} \cdot a_{i,j}^s \leq \dot{v}_{i,j}^s &\leq \dot{V}_{i,j}^{max} \cdot a_{i,j}^s && \forall s \in S,\ (i,j) \in GV(E) && (4.12a)\\
\Delta h_{i,j}^s &= 0 && \forall s \in S,\ (i,j) \in GV(E) && (4.12b)\\
p_{i,j}^s &= 0 && \forall s \in S,\ (i,j) \in GV(E) && (4.12c)
\end{aligned}
$$

5 Algorithmics for the System Design of Fluid Systems

While the optimization model for fluid systems was presented in Chapter 4, simply generating a corresponding MILP instance and trying to solve it using a standard MILP solver limits the applicability in practice. Since instances of real-world applications oftentimes tend to grow quickly, they often cannot be solved in reasonable time. An example are the instances considered later in this chapter. Standard MILP solvers fail to solve these instances in reasonable time because of the inability to provide strong dual bounds. Therefore, a contribution to the algorithmic synthesis of fluid systems on a larger scale, i.e. an algorithmic approach for solving practice-oriented instances, is presented in this chapter. The goal is to generate "good" systems in reasonable time. Here, "good" refers to solutions with a desirable value of the objective function. The focus is on the time aspect as the runtime is essential for the practical applicability. In this context, a domain-specific approach is used that exploits the special characteristics of technical systems by primal and dual heuristics. In order to ensure a certain practical relevance, the application example of so-called booster stations is examined. According to the principles of Algorithm Engineering, as explained in Sanders (2009), the orientation towards practice-oriented problems is an important feature since applications play an important role for the development of algorithms and serve as realistic inputs for meaningful experiments. This is especially the case if not all future applications for the algorithms are known in advance and hence providing algorithms validated on related applications with realistic inputs is an important factor (Sanders, 2009).

The basic idea for the presented approach is as follows: Use both the MILP and the graph representation of a problem simultaneously and benefit from both. On the primal side, a local search algorithm is used to obtain good primal solutions. In this thesis, the focus is on Simulated Annealing, but other local search algorithms, e.g. Genetic Algorithms or Tabu Search, are possible, too. In the context of Simulated Annealing, the graph representation is used to define neighborhoods and the MILP representation is used to evaluate the quality of the generated systems. On the dual side, a heuristic based on domain-specific and technical knowledge is used to relax the original problem. This allows to identify strong dual bounds. Finally, both heuristics are combined in a Branch-and-Bound framework to further strengthen the bounds and close the optimality gap between the primal and dual solutions. Thus, it is possible to obtain provably optimal solutions for the system design.

5.1 Application to Booster Stations

A booster station, also referred to as pressure booster system, is a network of either one type or different types of one or more connected pumps. A main field of application is the supply of whole buildings or higher floors, especially in high-rise buildings, with drinking water if the supply pressure provided by the water company is not high enough to meet the demand at all times. Therefore, booster stations are typically designed for maximum flow taking into account maximum pressure increase (European Commission, 2018). In general, a distinction between three different concepts can be made. These concepts are booster stations with cascade control, with continuously variable speed control of one pump, and with continuously variable speed control of all pumps. According to a report prepared for the European Commission (2018), the use of variable-speed drives is preferable because loads can vary considerably over time and the efficiency of a booster station massively depends on the ability to adapt to partial loads. In this thesis, the focus is therefore on the third concept, booster stations with continuously variable speed control of all pumps. Furthermore, it is assumed that the booster stations under consideration are directly connected to the water supply. If necessary, so-called normal zones are implemented, which can be served by the supply pressure itself and are therefore not connected to a booster station. This can be used to avoid overpressure for lower floors. For all other floors, overpressure can be avoided by installing pressure reducing valves if necessary.

The overall system consists of different types of components, such as pumps, pipes, and valves. In the following, the focus is on the pumps of the booster station and the other components are considered implicitly. Hence, the representation is simplified to a switchable interconnection of pumps that form a connected network with one source and one sink. The relevant physical variables are: the volume flow through the pumps, the pressure increase (or equivalently the head) generated by the pumps, their power consumption, and their rotational speed.

Since there was no suitable library of test instances for the considered area of booster station optimization, new test instances had to be designed in order to be able to validate the developed approach presented later in this chapter. To obtain different test instances, certain characteristics were varied and combined: the height of the considered building, the intended use of the building with the corresponding load profile, the hot water generation, and the available pumps. The names of the instances can be derived from the abbreviations for the respective characteristics.

Building height: There are two different fictional buildings. Both are high-rise buildings but vary in the number of floors. The first building ($B15$) is 15 floors high and the second building ($B10$) is 10 floors high. This means that different pressure increases and maximum volume flows are required as the building's height effects the pressure losses and demanded volume flows.

Intended use: The buildings are either used as a hypothetical hospital (H), a residential (R) or an office building (O). All usage types differ regarding their furnishing and consumption behavior. Hence, different maximum volume flows, pressure losses and load profiles occur. Depending on the usage, four or five load scenarios are distinguished.

Hot water generation: The generation of hot water is either centralized (C) or decentralized (D). This results in different pressure losses and required volume flows.

Available pumps: For each test instance, one of two disjoint sets of pumps with five variable-speed pumps each is available. All of them are based on the Wilo-Economy MHIE[1] model series, see Figure 5.1. The first set includes the pump types 203 to 403 of the model series (1) and the second set the pump types 404 to 1602 (2) with different prices and operational characteristics.

Given that the maximum volume flows and piping of the surrounding system are known a priori, the required pressure increase to be provided by the booster station can be determined by preprocessing the pressure losses of the surrounding system and assuming negligible pressure losses for the booster station itself. Table 5.1

[1] https://wilo.com/ie/en/Products-and-expertise/Series-Finder/Wilo-Economy-MHIE_110.html (accessed May 07, 2021)

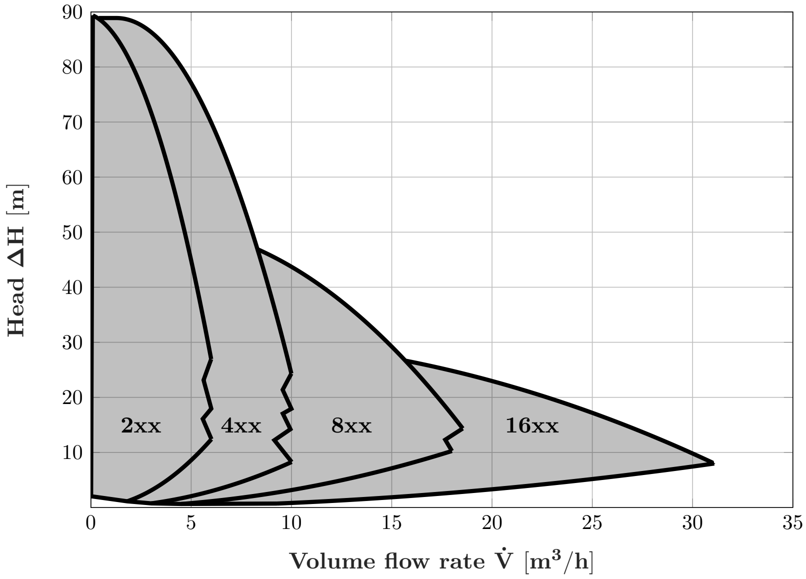


Figure 5.1: Schematic representation of the characteristic maps of the available pumps (based on Wilo SE, 2020a)

shows the peak loads for the different test instances in terms of the maximum volume flow $\dot{V}^{max}$ and required head ΔH. The head can also be expressed in terms of required pressure increase Δp by using the conversion $\Delta p = \Delta H \cdot \rho \cdot g$, with constant density ρ and acceleration due to gravity g. Furthermore, there are always two test instances for each of the 12 entries since two different pump kits are used, i.e. 24 test instances in total.

Table 5.1: Summary of the maximum volume flows $\dot{V}^{max}$ and required heads ΔH to be provided by the booster station for the considered test instances

Test instance	$\dot{V}^{max}$ $[m^3/h]$	ΔH $[m]$
B10_H_C_1/2	9.68	23.8
B10_H_D_1/2	14.53	31.8
B10_O_C_1/2	4.86	19.5
B10_O_D_1/2	6.48	27.5
B10_R_C_1/2	6.90	24.1
B10_R_D_1/2	9.85	32.1
B15_H_C_1/2	8.87	40.4
B15_H_D_1/2	10.49	48.4
B15_O_C_1/2	5.37	33.4
B15_O_D_1/2	6.45	41.4
B15_R_C_1/2	6.70	37.0
B15_R_D_1/2	7.92	45.0

For the partial loads, which depend on the considered building usage, Table 5.2 shows the different load scenarios with the relative time share F of the operational lifetime for which these scenarios are expected to occur and the associated relative volume flows $\dot{V}/\dot{V}^{max}$. Furthermore, the lifetime of the system is assumed to be ten years with mean energy costs of €0.30 per kWh for all test instances.

Table 5.2: Summary of the time shares F and relative volume flows $\dot{V}/\dot{V}^{max}$ of the different load scenarios depending on the building usage

Building	Scenario	F	$\dot{V}/\dot{V}^{max}$
Hospital	1	0.08	1.00
	2	0.25	0.50
	3	0.25	0.25
	4	0.17	0.10
	5	0.25	0.05
Office	1	0.03	1.00
	2	0.12	0.40
	3	0.30	0.30
	4	0.55	0.00
Residential	1	0.17	1.00
	2	0.17	0.70
	3	0.25	0.40
	4	0.21	0.35
	5	0.20	0.15

5.2 Primal Heuristic: Simulated Annealing

The implemented Simulated Annealing algorithm follows Boussaïd et al. (2013) with some modifications: Previous calculations are saved and a penalty term for non-valid system topologies is implemented. The algorithm is used to find good topologies for the first stage of the two-stage optimization problem described in Chapter 4. After generating a topology, the binary first stage variables are fixed in the MILP. Afterwards, the second stage is solved optimally for the chosen topology with respect to the different load scenarios using a standard solver.

For the topology decision, only (two-terminal) series-parallel networks are considered. These networks were first studied by MacMahon (1890) in the context of electrical networks, which show structural similarities with the fluid networks studied here. Therefore, they are briefly introduced in the following before continuing with the description of Simulated Annealing. Although series-parallel networks represent only a subset of all possible network structures, they are of great interest due to their simplicity (Riordan and Shannon, 1942). This simplicity is exploited to ensure that only technically sound systems are generated. For example, it is ensured that each technical component has at least one successor and one predecessor in the network. Following the definition provided by Booth and Tarjan (1993), a directed (multi-)graph $G = (V, E)$ with vertices V and edges E, where each directed edge $e \in E$ has a source $s(e) \in V$ and a sink $t(e) \in V$, is (two-terminal) series-parallel with source s and sink t if it can be generated using the following three construction rules:

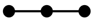
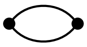
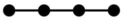
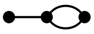
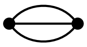
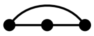
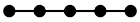
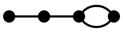
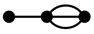
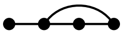
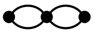
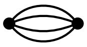
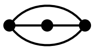
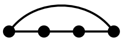
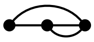
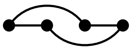
1. Any base graph $G = (\{s, t\}, \{e\})$ is series-parallel.
2. The graph G_P resulting from the parallel connection of two series-parallel graphs G_1 and G_2 by identifying s_1 with s_2 as source and t_1 with t_2 as sink is series-parallel.
3. The graph G_S resulting from the serial connection of two series-parallel graphs G_1 and G_2 by identifying t_1 with s_2 is series-parallel, with source s_1 and sink t_2.

To illustrate the concept, all series-parallel networks for two to four elements are shown in Table 5.3. For only one element, the resulting series-parallel network is the base graph itself. The networks are divided into so-called essentially series and essentially parallel networks. In this context, Riordan and Shannon (1942) define a network as essentially series or essentially parallel if it is created by the serial or parallel connection of two series-parallel networks. Using this classification, there is always the same number of essentially series and essentially parallel networks and for each network with a given number of elements of one class there is a corresponding network of the other class with the same number of elements that can be created by swapping the words series and parallel in the respective description. Two networks are considered to be equivalent if these can be obtained by the series or parallel connection of the same two networks (Riordan and Shannon, 1942).

Continuing the description of the implemented Simulated Annealing algorithm, the problem-specific neighborhood function consists of four single neighborhood operators, similar to those used in Altherr (2016). Exemplary illustrations of the respective neighborhoods are shown in Figure 5.2. These are the replace ($N_{Replace}$), the swap (N_{Swap}), the add (N_{Add}), and the delete neighborhood (N_{Delete}):

$$N = N_{Replace} \cup N_{Swap} \cup N_{Add} \cup N_{Delete}$$

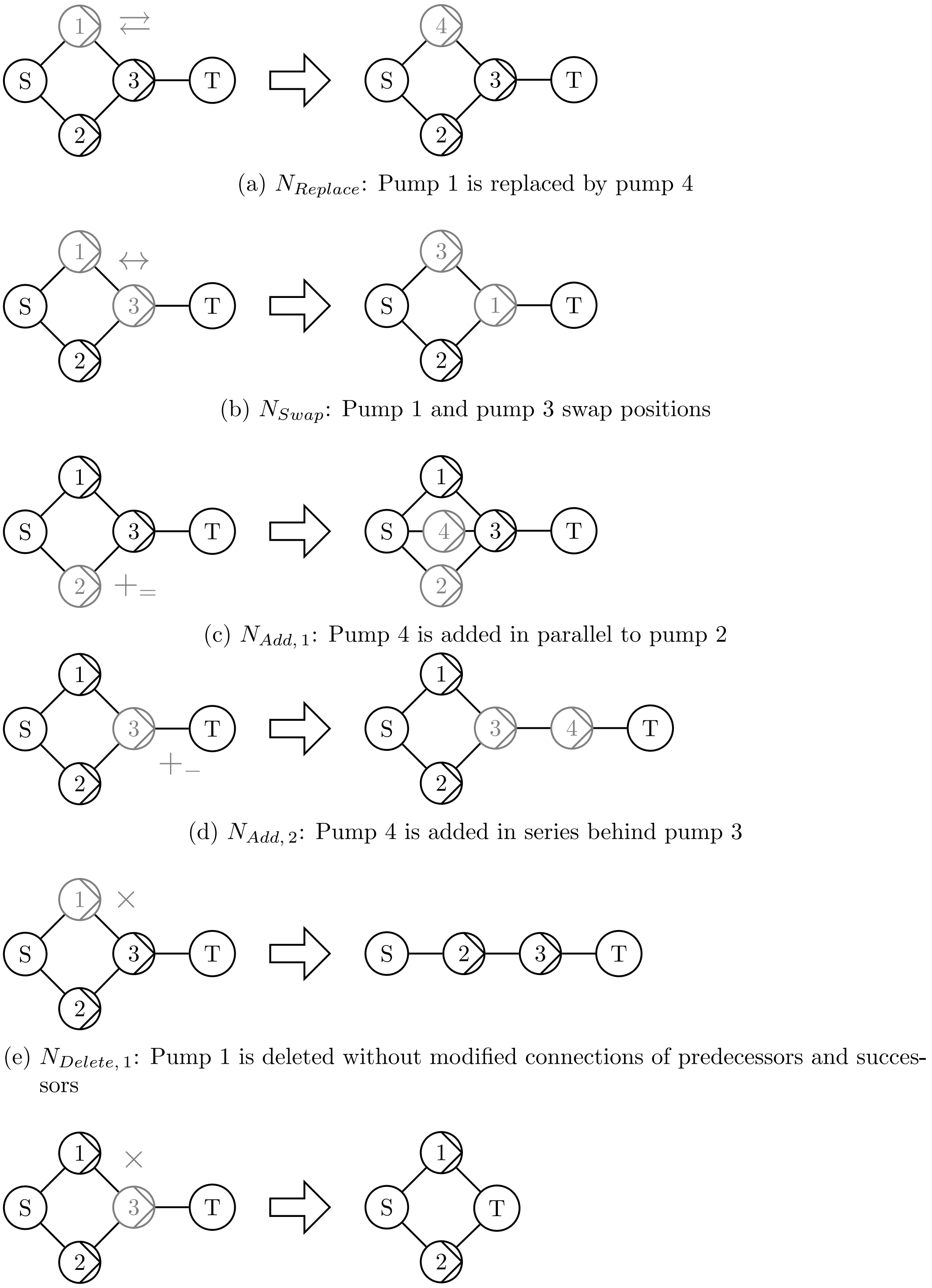
Table 5.3: Enumeration of the structures of all series-parallel networks with two to four elements (based on Riordan and Shannon, 1942)

#Elements	Essentially series	Essentially parallel	#Networks
2			2
3	 	 	4
4	    	    	10

Replace neighborhood: A component p_i, in the case of booster stations a pump, of the set of bought components—a subset of the set of available components—is selected randomly and replaced by a component p_j from the set of unbought components. The previous predecessors and successors of p_i are the new predecessors and successors of p_j. This neighborhood can only be created if the network consists of at least one component and there is at least one unbought component.

Swap neighborhood: Two different components p_i and p_j of the set of bought components are selected randomly. Components p_i and p_j swap positions in the network. The previous predecessors and successors of p_i are the new predecessors and successors of p_j and vice versa. This neighborhood can only be created if the network of bought components consists of at least two components.

Add neighborhood: A component p_i of the set of unbought components is selected randomly and it is decided whether p_i is connected in series or in parallel. If p_i is to be connected in series, a component out of the set of bought components, a source or a sink is selected. If a source or a sink is selected, p_i is connected in series behind the source or before the sink. If a component p_j is selected, p_i is connected before or behind p_j. The source, the sink or p_j becomes the new predecessor



(a) $N_{Replace}$: Pump 1 is replaced by pump 4

(b) N_{Swap}: Pump 1 and pump 3 swap positions

(c) $N_{Add,1}$: Pump 4 is added in parallel to pump 2

(d) $N_{Add,2}$: Pump 4 is added in series behind pump 3

(e) $N_{Delete,1}$: Pump 1 is deleted without modified connections of predecessors and successors

(f) $N_{Delete,2}$: Pump 3 is deleted with modified connections of predecessors and successors

Figure 5.2: Illustration of neighborhoods for the Simulated Annealing algorithm

or the new successor of p_i. Furthermore, p_i adopts the previous successors or predecessors. If p_i is to be connected in parallel, a component p_j of the set of bought components is selected. All predecessors and successors of p_j become the predecessors and successors of p_i as well. This neighborhood can only be created if the set of unbought components consists of at least one component and in the case of a parallel connection if the set of bought components consists of at least one component.

Delete neighborhood: A component p_i of the set of bought components is selected randomly and is deleted from the network. If a predecessor $p_{i,p}$ or a successor $p_{i,s}$ of p_i only has p_i as its successor or predecessor, a successor or predecessor of p_i is selected randomly. It then becomes the new successor or predecessor of $p_{i,p}$ or $p_{i,s}$. This is necessary to ensure flow conservation. Otherwise, the connection is deleted without substitution. This neighborhood can only be created if there is at least one component in the set of bought components.

To generate a starting solution, a simple heuristic based on N_{Add} is used to obtain valid solutions. First, a minimal network is considered. If this network is already a valid solution, it is accepted as the starting solution. Otherwise, components are added until a valid topology is generated. If the set of unbought components is empty and the solution is still not valid, the whole network is deleted and the procedure starts again with a minimal network until a valid solution is found.

For the considered problem, non-valid solutions have no associated costs. If the costs were set to $+\infty$, the algorithm would never accept them as the current solution. In this case, it would not be possible to reach every solution in the solution space with the defined neighborhood function. To avoid this, a penalty term is introduced assigning costs to non-valid solutions. If a solution is non-valid, double the costs of the starting solution are used instead. This approach has two advantages: First, the costs are low enough that non-valid solutions can be used as the current solution in the algorithm and second, high enough that they should be greater than the costs of all valid solutions.

The critical steps for the runtime of the algorithm are the calculations of the optimal operation performed by the MILP solver. To enhance the runtime of the algorithm, a list is created that holds the last calculated solutions. Each time a calculation is needed, the list is checked first whether this topology has already been calculated. If not, the solution is added to the list. If the list reaches the defined maximum size, the oldest entry is deleted to store new solutions.

Finally, a cooling schedule has to be determined. For this, an exponential cooling function $T(t) = T_0 \cdot \alpha^t$ is used, which is a common approach in related literature (Boussaïd et al., 2013). Here, T_0 is the starting temperature and t indicates the number of temperature reductions performed. The parameter α is a value between 0 and 1. It influences the slope of the cooling function. A threshold value T_{stop} acts as a termination criterion. As soon as the temperature falls below this threshold value, the algorithm terminates. Furthermore, the number of iterations per temperature level has to be chosen in such a way that the search space is explored sufficiently. These parameters have to be determined experimentally depending on the specific problem. In experiments, a value of $\alpha = 0.9$ showed good results in terms of ensuring a balance between runtime and exploration of the search space. The start temperature T_0 was set to 10 000. For the considered instances, especially with regard to the dimensions of the occurring costs, this proved particularly suitable to ensure both sufficient diversification and intensification. With regard to the dimension of expected costs, T_{stop} was set to 10. Hence, at the end of the algorithm, cost improvements were almost exclusively accepted in order to ensure intensification. To establish a balance at each temperature level, 100 iterations were carried out per temperature level. This proved to be favorable to explore the search space. At lower values the search space was reduced too much and at higher values the algorithm started to cycle excessively.

5.3 Dual Heuristic: Domain-Specific Relaxation

A simple LP-relaxation, i.e. dropping the integrality constraints, is not suitable to obtain strong lower bounds for the considered instances. For that reason, a special approach was developed that uses domain-specific knowledge in order to provide strong dual bounds, see Algorithm 1.

In the first step, the original problem is relaxed by disabling the coupling constraints that, in the original problem, connect the buy- ($b_{i,j}$) and the activation-variables ($a^s_{i,j}$) of the components for all load scenarios and serve to ensure that only bought components can be used:

$$a^s_{i,j} \leq b_{i,j} \qquad \forall\, s \in S, (i,j) \in E \tag{5.1}$$

Algorithm 1 Dual heuristic

```
Let $P_0$ be the original problem
Let $f$ be the objective function
Disable coupling constraints for $P_0$
Split $P_0$ into $P_1, ..., P_{|S|}$ (one for each load scenario $s$)
Lower Bound $LB \leftarrow 0$
Energy-Costs $EC \leftarrow 0$
Investment $I \leftarrow 0$
for $k \in \{1...|S|\}$ do
    Replace buy- with activation-variables for $P_k$
    Generate topology problem $T_k$ for $P_k$
    Generate operation problem $C_k$ for $P_k$
    $EC \leftarrow EC + f(T_k)$
    if $I < f(C_k)$ then
        $I \leftarrow f(C_k)$
    end if
end for
$LB \leftarrow EC + I$
```

In the case of booster stations, as considered here, the term “components” corresponds to pumps. After disabling the coupling constraints, the problem is split into $|S|$-many subproblems, one for each considered load scenario. The buy-variables in all subproblems are substituted by the corresponding activation-variables. Subsequently, each of the $|S|$ subproblems is split again into two sub-subproblems. The respective problems represent the optimization tasks of minimizing the energy costs (operation problem) and minimizing the investment (topology problem) for one single load scenario. The new objective functions for the sub-subproblems are:

$$\min \quad C^{kWh} \cdot OLT \cdot F^s \cdot \sum_{(i,j)\in E} p^s_{i,j} \qquad \forall\, s \in S \tag{5.2}$$

$$\min \quad \sum_{(i,j)\in E} \left(C^{buy}_{i,j} \cdot a^s_{i,j} \right) \qquad \forall\, s \in S \tag{5.3}$$

For each of these $2 \cdot |S|$ problems, the optimal solution is determined by a MILP solver. Using the solutions of these problems, a lower bound for the original problem can be obtained. This lower bound is composed of the sum of the energy costs and the maximum of the individual investments for all load scenarios:

$$\underline{z} = C^{kWh} \cdot OLT \cdot \sum_{s\in S} \left(F^s \cdot \sum_{(i,j)\in E} p^s_{i,j} \right) + \max_{s\in S} \left(\sum_{(i,j)\in E} \left(C^{buy}_{i,j} \cdot a^s_{i,j} \right) \right) \tag{5.4}$$

This is obviously a valid way to obtain lower bounds: The energy costs for a load scenario cannot be lower than the energy costs that arise for the decoupled case because this is also the configuration with minimal costs for the original problem in the given load scenario. Therefore, the sum of these energy costs cannot be higher than the total energy costs for the original problem. Given the fact that the optimal system for the original problem must be able to operate in each load scenario, the investment cannot be lower than the maximum of the individually computed investments for each decoupled load scenario because there is no cheaper configuration to cover the "most challenging" load scenario.

5.4 Closing the Gap: Branch-and-Bound

Based on the basic Branch-and-Bound method, as described in Clausen (1999), a framework using domain-specific knowledge to obtain optimal solutions for the considered minimization problem is presented, see Algorithm 2. Branch-and-Bound belongs to the class of exact solution methods. It is a widespread method for solving large, combinatorial optimization problems. The complete enumeration of such problems is impractical because the number of possible solutions grows exponentially with the problem size. However, the advantage of the Branch-and-Bound method is that parts of the solution space can be pruned. For this, a dynamically generated search tree is used.

Initially, this search tree only consists of one node, the root node, which represents the whole search space of the original problem. Typically, a feasible solution for the root problem is calculated in advance and becomes the initial best known solution, where the best known solution value is a synonym for the global upper bound. If no feasible solution is calculated in advance, the best known solution value is set to $+\infty$ when considering a minimization problem. Here, the solution of the Simulated Annealing implementation described in Section 5.2 is used for this purpose.

In each iteration of the Branch-and-Bound method, an unexplored (active) node, representing a specific subproblem, is processed. In the presented implementation, an active node implies that one or more so-called conflicting components exist in the solution of the relaxation for this subproblem. These are components that are used for operation in the relaxation, but whose purchase costs are not part of the associated investment. An iteration contains three steps: selecting a node, dividing the solution space of this node into two smaller subspaces (branching) and calculating the bounds for the arising subproblems.

Algorithm 2 Branch-and-Bound framework

```
Calculate global Upper Bound $UB$
Let $n_0$ be the original problem
Calculate Lower Bound $LB(n_0)$ of $n_0$
Let $RELAX(n_0)$ be the optimal solution for the relaxation of $n_0$
Active Nodes $AN \leftarrow AN \cup \{n_0\}$
while $AN \neq \emptyset$ do
    Take $n_i \in AN$ with $LB(n_i) \leq LB(n_j)\ \forall n_j \in AN$
    Take a component $p_p$ from $n_i$ that is used but not bought
    Split $n_i$ into two child nodes $cn_0, cn_1$
    Fix buy-variable of $p_p$ to 0 for $cn_0$ and to 1 for $cn_1$
    for $cn_k \in \{cn_0, cn_1\}$ do
        $AN \leftarrow AN \cup \{cn_k\}$
        Calculate $LB(cn_k)$
        if $LB(cn_k) \geq UB$ then
            $AN \leftarrow AN \backslash \{cn_k\}$
        else if $RELAX(cn_k)$ is a valid solution for $n_0$ then
            $AN \leftarrow AN \backslash \{cn_k\}$
            $UB \leftarrow LB(cn_k)$
        end if
    end for
    $AN \leftarrow AN \backslash \{n_i\}$
end while
Optimal Solution $OS \leftarrow UB$
```

The selection of a node follows a certain selection strategy. Here, the best-first-search selection strategy is used, where always the node out of the set of active nodes with the lowest bound is selected.

After the selection, branching is performed and two child nodes are generated by introducing additional constraints in order to divide the solution space. The branching rule for the active nodes is defined as follows: A component out of the set of conflicting components of this node is selected randomly. For one of the subproblems, an additional constraint is added that sets the binary buy-variable of the selected conflicting component to zero, i.e. the component is not part of the system. For the other subproblem, an additional constraint that sets the buy-variable to one, i.e. the component is part of the system, is added instead. Hence, the search space is split into two smaller disjoint search spaces. If a buy-variable is set to zero, the selected conflicting component is not bought and therefore cannot be used for operation. As a result, any solution with an activation-variable associated with this component not equal to zero would be inherently infeasible for the original problem, see the relaxed Constraint (5.1). Therefore, the activation-variables associated with these components in the respective subproblems are fixed to zero. In the opposite case, the activation-variables are not affected by such a restriction.

Afterwards, the bounds of the newly generated nodes are calculated immediately. This is called the eager evaluation strategy, whereas for the so-called lazy strategy, the bounds of the child nodes are not calculated until the respective node is selected and the node selection would be based on the bound of the respective parent node. In the implementation presented here, the bound of a node is determined according to Equation (5.4) by solving the relaxation for the given subproblem. If the solution of the relaxation of a node is a valid solution for the original problem, its value is compared to the currently best known solution value and the better solution is kept. In this context, a solution of the relaxation is valid for the original problem if and only if all components used for operation are also bought, i.e. there are no conflicting components. If the bound is worse than the best known solution value, no further exploration of this subtree is needed because the subproblem contains no better solutions for the original problem than the currently best known solution. The same applies if there are no feasible solutions for the subproblem. Otherwise, if none of these three cases occur, the node becomes part of the set of active nodes since the corresponding subproblem might still contain better solutions than the currently best known solution.

The search ends if there are no active nodes left. The currently best known solution at this point is the provably optimal solution to the original problem since there are no subproblems that could contain a better solution and the union of their disjoint search spaces equals the search space of the original problem.

An exemplary illustration for branching in the case of the application to booster stations is given in Figure 5.3. The procedure starts from the root node N_0 with initial best known solution z_{best} resulting from using the objective value of the solution produced by Simulated Annealing z_{SA}. Branching is performed on the buy-variables of the conflicting pumps as described above, here represented by b_{Px}. The node indices indicate the sequence of the node creation. Furthermore, the example includes all three termination criteria for a node: the solution of the relaxation is also a valid solution for the original problem (N_3 & N_7), the subproblem is infeasible (N_4 & N_8) or the bound obtained by the relaxation $\underline{z}_x$ is worse than the currently best known solution (N_5).

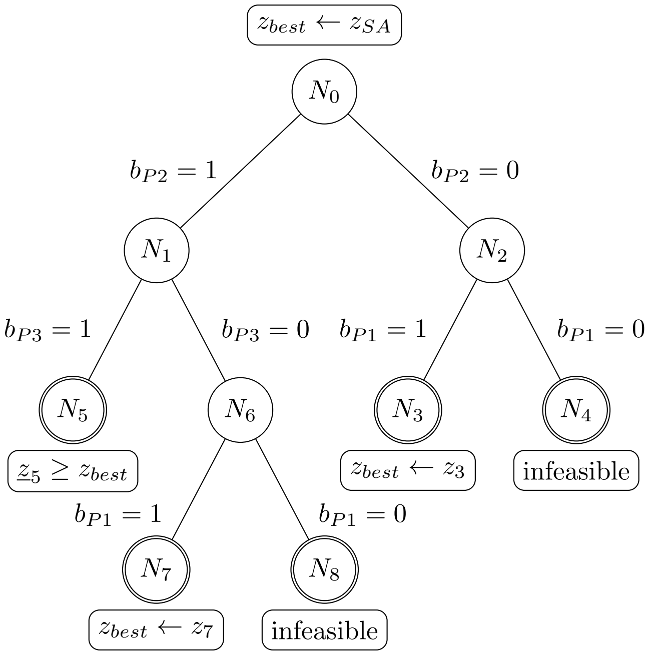


Figure 5.3: Illustration of the implemented Branch-and-Bound framework

5.5 Computational Study

In order to validate the developed approach, a computational study using the 24 test instances introduced in Section 5.1 was conducted. All calculations were performed on a MacBook Pro Early 2015 with a 2.7 GHz Intel Core $i5$ and 8 GB 1867 MHz DDR3 memory, using CPLEX Optimization Studio 12.6 as MILP solver.

5.5.1 Examination of the Solution Quality

In this section, the quality of the solutions found by the implemented Simulated Annealing algorithm and the domain-specific dual heuristic is discussed.

Simulated Annealing: An overview of the obtained results in terms of solution quality for the presented implementation of Simulated Annealing in all test instances is shown in Table 5.4. The best solutions found by Simulated Annealing are represented by z_{SA}. First, these solutions are compared to the corresponding lower (dual) bounds $\underline{z}$ calculated using the proposed dual heuristic. The relative gap between the solution of Simulated Annealing and the dual bound $gap_{\underline{z}}$, which is typically used to evaluate heuristically obtained solutions in practice since optimal solutions are usually not known, is defined as $(z_{SA} - \underline{z})/\underline{z}$. The mean value of $gap_{\underline{z}}$

for all test instances was 9.27% with a standard deviation of 6.37%. However, since the actual optimal solutions, denoted by z^*, are known for the test instances under consideration, the solutions found by Simulated Annealing are then compared to the optimal solutions obtained via Branch-and-Bound. The relative gap between the best solution found by Simulated Annealing and the actual optimal solution gap_{z^*} is defined as $(z_{SA} – z^*)/z^*$. In 14 out of 24 cases, the optimal solution was found by the implemented Simulated Annealing algorithm. The mean value of gap_{z^*} was 0.69% with a standard deviation of 1.08%. However, if the optimal solution was not found, the mean value of gap_{z^*} was still only 1.65% with a standard deviation of 1.1%. These results show that the implemented Simulated Annealing algorithm is able to find favorable systems with low lifetime costs for the considered application. A more detailed examination with regard to the topology of these systems is given in Section 5.5.2.

Table 5.4: Overview of the results for the implemented Simulated Annealing algorithm with regard to the dual bounds and optimal solutions

Test instance	z_{SA} [€]	$\underline{z}$ [€]	$gap_{\underline{z}}$ [%]	z^* [€]	gap_{z^*} [%]
B10_H_C_1	13 315.10	12 659.00	5.18	13 070.80	1.87
B10_H_C_2	13 946.80	11 168.80	24.87	13 946.80	0.00
B10_H_D_1	25 068.10	23 912.40	4.83	24 607.60	1.87
B10_H_D_2	23 704.70	22 127.50	7.13	23 287.70	1.79
B10_O_C_1	4 370.36	4 024.15	8.60	4 370.36	0.00
B10_O_C_2	4 712.02	4 224.70	11.54	4 712.02	0.00
B10_O_D_1	6 007.54	5 962.32	0.76	6 007.54	0.00
B10_O_D_2	6 492.46	6 026.12	7.74	6 492.46	0.00
B10_R_C_1	12 711.90	12 334.20	3.06	12 711.90	0.00
B10_R_C_2	13 968.60	12 157.00	14.90	13 968.60	0.00
B10_R_D_1	24 601.00	24 004.30	2.49	24 518.10	0.34
B10_R_D_2	23 516.20	22 215.40	5.86	23 516.20	0.00
B15_H_C_1	21 380.80	18 942.90	12.87	20 974.60	1.94
B15_H_C_2	21 649.00	17 637.20	22.75	21 041.10	2.89
B15_H_D_1	27 936.80	26 651.30	4.82	27 210.70	2.67
B15_H_D_2	28 186.40	25 001.00	12.74	27 377.30	2.96
B15_O_C_1	6 571.15	6 162.81	6.63	6 571.15	0.00
B15_O_C_2	7 002.53	6 288.44	11.36	7 002.53	0.00
B15_O_D_1	10 069.90	10 015.90	0.54	10 069.90	0.00
B15_O_D_2	9 115.01	8 116.26	12.31	9 115.01	0.00
B15_R_C_1	20 505.40	19 750.90	3.82	20 486.40	0.09
B15_R_C_2	19 909.10	17 315.50	14.98	19 909.10	0.00
B15_R_D_1	29 360.00	27 570.20	6.49	29 319.40	0.14
B15_R_D_2	28 407.00	24 457.10	16.15	28 407.00	0.00

Lower Bounds: In order to validate the quality of the lower bounds, these are compared to the optimal solution. Table 5.5 summarizes the results. Again, $\underline{z}$ represents the lower bounds for the respective test instances and z^* the optimal solutions obtained using Branch-and-Bound. The relative gap between the lower bounds and the optimal solutions, denoted by *gap*, is defined as $(z^* - \underline{z})/\underline{z}$. The mean value of *gap* was 8.52% with a standard deviation of 6.28%. The maximum value of *gap* was 24.87%, while the minimum was only 0.54%. Overall, these results indicate that the developed domain-specific dual heuristic is capable of providing strong lower bounds for the considered application.

Table 5.5: Overview of the results for the implemented dual heuristic with regard to the optimal solutions

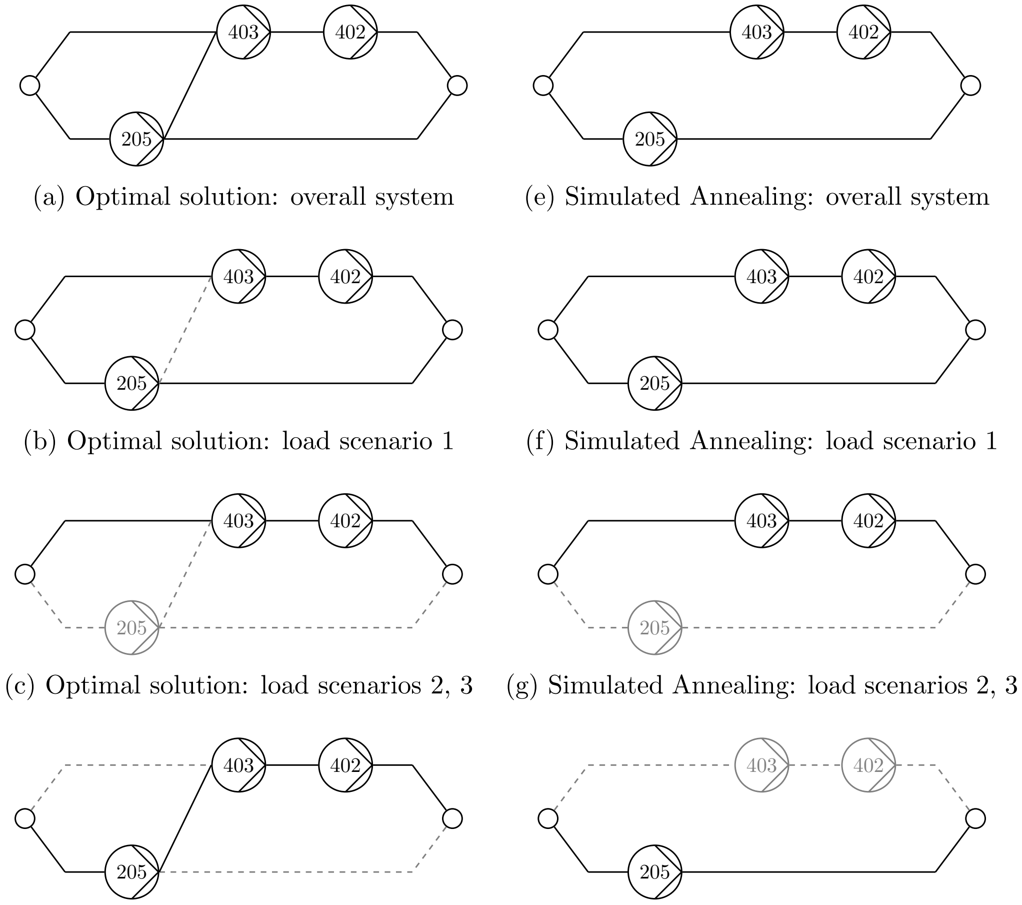
Test instance	$\underline{z}$ [€]	z^* [€]	*gap* [%]
B10_H_C_1	12 659.00	13 070.80	3.25
B10_H_C_2	11 168.80	13 946.80	24.87
B10_H_D_1	23 912.40	24 607.60	2.91
B10_H_D_2	22 127.50	23 287.70	5.24
B10_O_C_1	4 024.15	4 370.36	8.60
B10_O_C_2	4 224.70	4 712.02	11.54
B10_O_D_1	5 962.32	6 007.54	0.76
B10_O_D_2	6 026.12	6 492.46	7.74
B10_R_C_1	12 334.20	12 711.90	3.06
B10_R_C_2	12 157.00	13 968.60	14.90
B10_R_D_1	24 004.30	24 518.10	2.14
B10_R_D_2	22 215.40	23 516.20	5.86
B15_H_C_1	18 942.90	20 974.60	10.73
B15_H_C_2	17 637.20	21 041.10	19.30
B15_H_D_1	26 651.30	27 210.70	2.10
B15_H_D_2	25 001.00	27 377.30	9.50
B15_O_C_1	6 162.81	6 571.15	6.63
B15_O_C_2	6 288.44	7 002.53	11.36
B15_O_D_1	10 015.90	10 069.90	0.54
B15_O_D_2	8 116.26	9 115.01	12.31
B15_R_C_1	19 750.90	20 486.40	3.72
B15_R_C_2	17 315.50	19 909.10	14.98
B15_R_D_1	27 570.20	29 319.40	6.34
B15_R_D_2	24 457.10	28 407.00	16.15

5.5.2 Examination of the Topologies

As described above, Simulated Annealing could not find the optimal system in all test instances. For this reason, test instance B15_H_D_1 is discussed in this section with regard to the different topologies between the overall system found by Simulated Annealing and the optimal system as well as the subsystems used in the different load scenarios. Subsequently, the findings are transferred to the other non-optimal systems obtained by using Simulated Annealing. A complete overview of all optimal systems and those obtained with Simulated Annealing, as well as the subsystems used in each load scenario, can be found in Appendix A.1.

To compare the topologies, the found variable assignments were retranslated into the corresponding overall systems and subsystems for the individual load scenarios. Figure 5.4a shows this for the optimal system topology of test instance B15_H_D_1. As can be seen, the optimal topology has a non-series-parallel structure since the 205-type pump is connected with the other two pumps both in parallel and in series. Due to the restriction of the system topology to series-parallel structures, this system cannot be found by the implemented Simulated Annealing. The series-parallel system found by Simulated Annealing is shown in Figure 5.4e. It is noticeable that this system is quite similar to the optimal system. The difference between both systems is that the 205-type pump in the series-parallel system is only connected in parallel to the other pumps and not additionally in series. Since the same pumps are installed in both systems, the investment is identical.

Although the overall system topology in the optimal case is not series-parallel, only series-parallel subsystems are used in the individual load scenarios. These are shown in Figures 5.4b, 5.4c and 5.4d for all five load scenarios. Figure 5.4b for the first load scenario and Figure 5.4d for the fourth and fifth load scenario show why a non-series-parallel structure is present for the optimal topology although only series-parallel subsystems are used in the individual load scenarios. In the first load scenario, the 205-type pump is operated in parallel and in the fourth and fifth load scenarios, it is operated in series with the other pumps. For comparison, Figures 5.4f, 5.4g and 5.4h show the subsystems of the series-parallel overall system found by Simulated Annealing that are used in the individual load scenarios. In the first load scenario, see Figure 5.4f, the same subsystem is used as for the optimal solution, see Figure 5.4b. This is also the overall system found by Simulated Annealing. In the second and third load scenario, see Figure 5.4g, the subsystems used in the solution found by Simulated Annealing also correspond to the subsystems used in the optimal system shown in Figure 5.4c. However, there are differences in the fourth and fifth load scenario. In these load scenarios, the serial connection of the 205-type pump with the other two pumps is missing in comparison to the optimal system. Instead of a serial connection of all three pumps, as shown in



(a) Optimal solution: overall system

(e) Simulated Annealing: overall system

(b) Optimal solution: load scenario 1

(f) Simulated Annealing: load scenario 1

(c) Optimal solution: load scenarios 2, 3

(g) Simulated Annealing: load scenarios 2, 3

(d) Optimal solution: load scenarios 4, 5

(h) Simulated Annealing: load scenarios 4, 5

Figure 5.4: Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B15_H_D_1

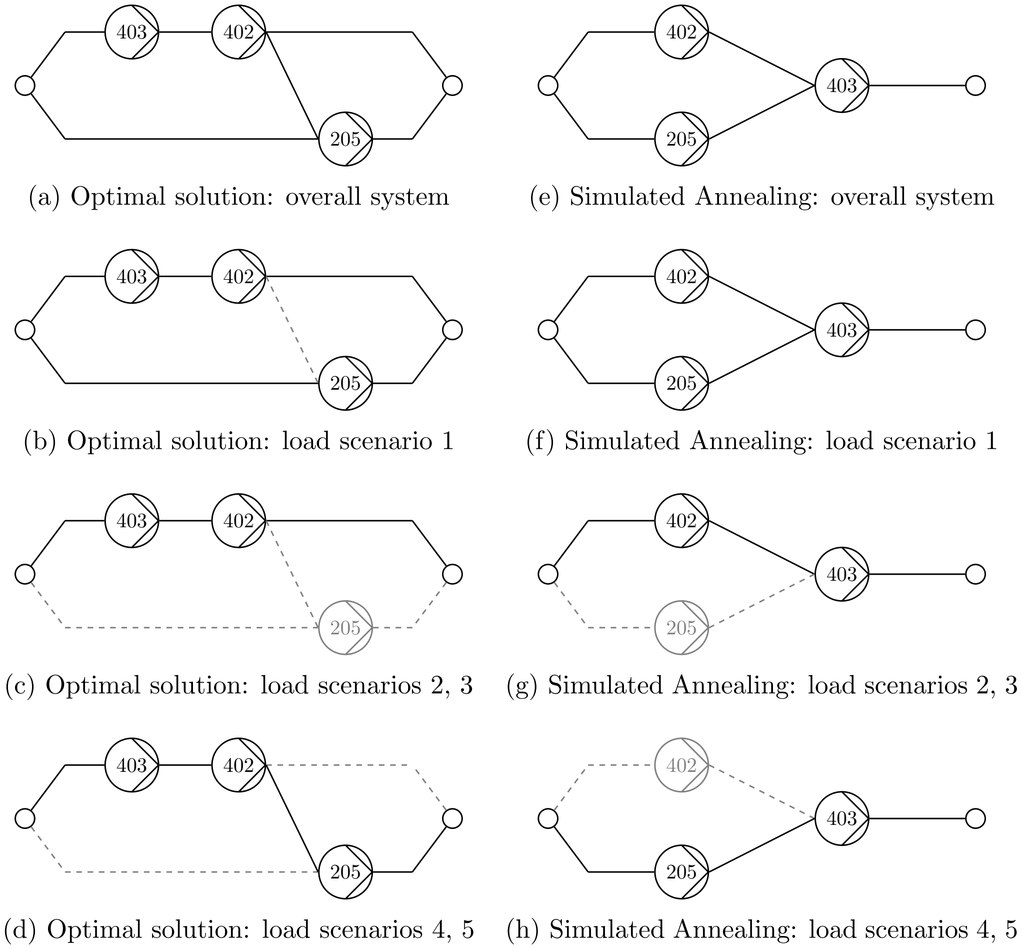
Figure 5.4d, only the 205-type pump is used. Thus, both configurations only differ in the operation in the fourth and fifth load scenario. Due to the low load in the fourth and fifth load scenario compared to the other load scenarios, the energy costs are only €726.10 higher over the estimated lifetime of ten years. Thus, in terms of total costs, only 2.67% higher costs arise.

In addition, the topologies of the remaining nine test instances for which Simulated Annealing was not able to find the optimal system were examined. In all cases, the optimal system has a non-series-parallel structure analogous to the discussed test instance. The non-series-parallel structures arise because pumps are interconnected both in series and in parallel. However, even in these cases only series-parallel subsystems are used in the individual load scenarios.

It is noticeable that the systems found by Simulated Annealing always have the same pumps as the optimal systems. Also, the interconnection of the pumps is, except for test instance B15_H_C_1, always a subsystem of the optimal system. It differs only by the connections that cause the overall system to become non-series-parallel. In addition, wherever the operation of the optimal subsystem for the individual load scenarios is possible due to the overall topology, the optimal subsystem for the respective load scenario is used, as expected. With the exception mentioned above, the system topologies of the systems found using Simulated Annealing correspond to the subsystem of the optimal system used in the "most challenging", i.e. the first, load scenario. Therefore, the subsystems used in the first load scenario are identical for both systems. This correlation also occurs for test instance B15_H_D_1 examined at the beginning of this section and can be seen in Figures 5.4b, 5.4e and 5.4f.

The system topologies for the aforementioned exception, test instance B15_H_C_1, are shown in Figures 5.5a and 5.5e. As described, these have the same pumps, but unlike the other test instances, the topologies differ more. This can have two possible causes: On the one hand, there may be a better series-parallel system that was not found by Simulated Annealing. On the other hand, the system found may be favorable in comparison to the subsystem used in the first load scenario for the optimal system, which is shown in Figure 5.5b. By comparing the costs of the two systems, it turned out that the system found by Simulated Annealing with total costs of €21 380.80 is 0.9% better than the series-parallel subsystem used in the optimal system to fulfill the "most challenging", first load scenario. This subsystem has total costs of €21 577.81. Thus, the test instance is indeed an exception compared to the other test instances.

In most cases, Simulated Annealing is able to find optimal systems if those systems have a low complexity and a comparatively small number of pumps and connections. The fact that no optimal solutions were found for more complex systems is not due to the implemented Simulated Annealing itself but due to the limitation to series-parallel systems. With increasing complexity, non-series-parallel systems generally gain in importance in terms of the number of possible systems compared to the series-parallel case. In addition, these systems have the advantage that installed pumps can be operated in parallel or in series in different load scenarios as required. This provides more flexibility to cover a certain load scenario as energy-



(a) Optimal solution: overall system (e) Simulated Annealing: overall system

(b) Optimal solution: load scenario 1 (f) Simulated Annealing: load scenario 1

(c) Optimal solution: load scenarios 2, 3 (g) Simulated Annealing: load scenarios 2, 3

(d) Optimal solution: load scenarios 4, 5 (h) Simulated Annealing: load scenarios 4, 5

Figure 5.5: Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B15_H_C_1

efficient as possible without adding further pumps to the system. However, this also results in a larger number of possible topologies with the same number of components. Hence, it becomes more difficult to create valid systems. Furthermore, the size of the search space increases. This in turn can have a negative effect on the quality of the found solutions and the runtime. Nevertheless, even within the search space restricted to series-parallel systems, good solutions could be found, whose objective function values deviated by less than 3% from those of the optimal solutions.

5.5.3 Examination of the Runtimes

In this section, the results for all three procedures are presented and briefly discussed in terms of runtime. An overview of the runtimes for Simulated Annealing (SA), the dual heuristic to generate initial lower bounds (LB) and Branch-and-Bound (B&B) in all test instances is given in Table 5.6. For the Branch-and-Bound framework, the best solutions found by Simulated Annealing were used as the starting solutions, i.e. initial upper bounds.

Table 5.6: Overview of the runtimes for the implemented Simulated Annealing algorithm, the dual heuristic and the Branch-and-Bound framework

Test instance	SA [s]	LB [s]	B&B [s]
B10_H_C_1	275	653	13 557
B10_H_C_2	109	574	9 956
B10_H_D_1	439	801	5 948
B10_H_D_2	417	742	13 615
B10_O_C_1	102	372	7 313
B10_O_C_2	85	208	4 148
B10_O_D_1	140	398	4 260
B10_O_D_2	143	260	4 996
B10_R_C_1	329	828	8 382
B10_R_C_2	333	687	21 473
B10_R_D_1	551	1 095	10 518
B10_R_D_2	652	1 582	12 522
B15_H_C_1	1 162	957	18 796
B15_H_C_2	424	583	12 039
B15_H_D_1	2 412	709	6 273
B15_H_D_2	660	707	13 243
B15_O_C_1	353	612	6 484
B15_O_C_2	188	269	5 249
B15_O_D_1	437	811	7 542
B15_O_D_2	224	310	4 848
B15_R_C_1	399	767	10 177
B15_R_C_2	231	426	12 361
B15_R_D_1	1 054	1 010	11 827
B15_R_D_2	290	501	13 721

Simulated Annealing: The implemented Simulated Annealing algorithm took on average 475 seconds to terminate. However, comparably high deviations occurred. The maximum runtime was 2 412 seconds, while the minimum runtime was only 85 seconds. This deviation results from the fact that the MILP solver needs much more time to solve the operation subproblem if the created neighborhood is large in terms of many bought components. Nevertheless, the runtimes are fast enough to be of practical relevance for the considered application.

Lower Bounds: Generating lower bounds took on average 661 seconds. The longest runtime was 1 582 seconds, while the shortest runtime was only 208 seconds. In most cases, this was comparable to the time the Simulated Annealing algorithm took to terminate. Hence, this circumstance allows a timely examination of a solution found by Simulated Annealing in practice.

Branch-and-Bound: The average runtime for obtaining optimal solutions was 9 969 seconds. The maximum runtime was 21 473 seconds and the minimum runtime only 4 148 seconds. If the initial upper bound found by Simulated Annealing was already the optimal solution, the average runtime was 8 804 seconds and therefore 31.75% less than in the opposite case, where the average runtime was 11 599 seconds. An overview of the solution progress for obtaining the optimal solutions as a function of time for all test instances can be found in Appendix A.2. Overall, the results show that with the proposed domain-specific Branch-and-Bound framework, it is possible to provide provably optimal solutions for instances on a practice-oriented scale in reasonable time.

6 Resilient System Design

Beyond the general system design under consideration, this chapter provides an outlook on how the resilience of technical systems in the considered setting can be enhanced by adding possible breakdown scenarios. The concept of resilience is of great interest since it cannot only be applied to control uncertainty during the design phase, but it is also applicable to the system's operation. Instead of designing systems that are robust with respect to specific single "what-if" assumptions made beforehand during the design phase, resilient system design aims at creating systems that perform "no matter what" (Altherr et al., 2018a). In this context, resilience of a technical system is the ability to overcome minor failures and thus to avoid a complete breakdown of its vital functions. A possible failure of the system's components is one critical case the system designer should keep in mind. For this, optimization under uncertainty can be used in order to describe and increase resilience of technical systems (Altherr et al., 2018a). The most prominent solution paradigms for optimization under uncertainty are Stochastic Programming (see e.g. Birge and Louveaux, 2011) and Robust Optimization (see e.g. Ben-Tal et al., 2009).

For the design of resilient technical systems, the following special case is considered: Starting from a valid network configuration (G, S_G, T_G) capable of covering the expected loads of any scenario $i \in S$, with graph $G = (V, E)$, edges E and vertices V, where S_G and T_G are the sources and sinks, it is allowed to add some additional components to make the system more resilient against breakdowns. More concrete, $I = E$ is defined as the set of initial components, A as the set of additional components and it is tried to find a subset $A' \subseteq A$ such that $G' = ((V, I \cup A'), S_G, T_G)$ fulfills resilience in the following sense: For each scenario $i \in S$, it has to be ensured that if a single component $e \in I$ fails, the remaining network $G'' = ((V, (I \cup A') \backslash \{e\}), S_G, T_G)$ suffices to cover the expected load in scenario i. The set of added additional components A' must be selected such that the lifetime costs of the resulting system, i.e. the sum of the investment and the operational costs, are minimal.

Since the system design process can be conducted in several consecutive steps, this leads to a multistage optimization problem with five stages: Design or adapt the system (1) such that for each anticipated load scenario (2) the optimal operation point can be found (3) and it can be ensured for each breakdown case (4) that the functionality of the system is maintained (5). A straightforward way to model multistage optimization problems are so called Quantified Mixed-Integer Linear Programs (QMIP). Hence, using QMIPs to find optimal system configurations with increased resilience is a suitable approach in the context of the applications under consideration. In this regard, this thesis focuses on the engineering perspective of the approach. It is intended to show how the approach can be applied in a practical setting and how it can be integrated into the design process of technical systems. For a more in-depth discussion regarding the formal and algorithmic properties, see Hartisch (2020).

Quantified Mixed-Integer Linear Programming is a direct and formal extension to Mixed-Integer Linear Programming. In QMIPs the variables are ordered explicitly and they are quantified either existentially or universally resulting in a multistage optimization problem under uncertainty. The objective function is an alternating minmax function according to the quantifier sequence: Existential variables are set with the goal of minimizing the objective value while obeying the constraint system, whereas universal variables are aiming at a maximized objective value. An instance of a QMIP can be visualized as a two-person zero-sum game between an existential player and a universal player, who alternately set the existentially and universally quantified variables (Hartisch, 2020). In the considered case, the existential player would be the system designer, while the universal player describes events that are by nature uncertain. Hence, solutions of QMIPs are strategies for assigning existentially quantified variables such that the linear constraint system is fulfilled. One way to deal with quantified programs is to build the corresponding Deterministic Equivalent Program (DEP) (see e.g. Wets, 1974; Wolf, 2015) and to solve the resulting MILP using standard MILP solvers.

6.1 Application to Booster Stations

In order to build on the results of Chapter 5, the application example of generating cost-optimal resilient booster stations out of non-resilient ones is examined here. Although applied to booster stations in this thesis, the approach can be abstracted for a variety of technical systems using the general representation of so-called process networks as shown in Hartisch et al. (2018).

The requirements for the considered resilient booster stations are manifested in DIN 1988-500. It specifies that booster stations must have at least one stand-by pump and that the system must be capable of covering the peak load and thus all required loads at all times in the event of a failure of a single pump. The relevant costs considered here for obtaining a resilient booster station are the investment for the stand-by pumps as well as the energy costs of the overall system in relation to the expected load scenarios and the operational lifetime of the system. The breakdown cases are expected to only take place in a small amount of time compared to the total lifetime due to short repair times. Therefore, the associated costs do not significantly affect the energy costs of the system and are neglected.

In contrast to related contributions (cf. Altherr et al., 2019), a further requirement mentioned in DIN 1988-500 is considered. This requirement states that in order to avoid stagnation water, an automatic, cyclic interchange between all pumps including the stand-by pumps is required. Hence, all pumps must be operated at least once every 24 hours. In order to meet this requirement, it is defined that the considered load scenarios represent a daily recurring load profile. Therefore, the requirement is assumed to be met if each pump is active in at least one load scenario. It is obvious that this additional requirement to use the stand-by pump in at least one load scenario has a massive impact on the operation of the system and thus also on its expected energy costs. Given this circumstance, it is not a trivial task to determine by which stand-by pumps the system should be extended in order to obtain a cost-optimal resilient system.

In general, single pumps or sets of pumps can be connected in parallel or in series. According to today's practice, parallel connections are state of the art due to a reduced overall complexity (Betz, 2017; Meck et al., 2020). As discussed in Betz (2017), there are further reasons to only consider parallel arrangements, although serial arrangements are generally conceivable: Firstly, control strategies for serial arrangements are more difficult to realize in practice. Secondly, in case of a failure of a single pump, the remaining system components are not directly affected and retain their full functionality, which is consistent with the considered definition of resilience. Therefore, only parallel arrangements are used in the following to demonstrate the approach, resulting in significantly smaller pump networks. An illustration of such a network with four pumps connected exclusively in parallel is shown schematically in Figure 6.1.

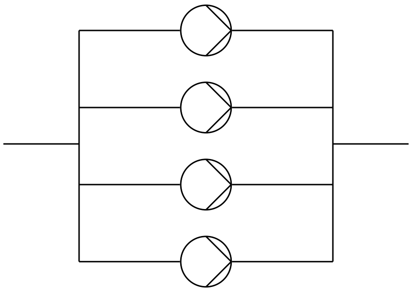


Figure 6.1: Illustration of a booster station with exclusively parallel pumps

6.2 Optimization Model

As outlined above, the quantified optimization model considered here has five stages. The first stage represents the investment decision to buy additional stand-by pumps. This stage is classified as existential. In the second, universal stage a load scenario is selected. In the subsequent existential stage, the optimal operation for this load scenario is determined with regard to the available pumps—the initial pumps as well as the additional stand-by pumps. These three stages represent the decisions during the "regular operation", i.e. operation without the failure of a pump, while the following two stages deal with the failure of a pump. In the fourth, universal stage, one of the initial pumps is selected and it is assumed that this pump is broken. At this point, only the initial pumps are considered for breakdown because if a stand-by pump fails, operation is already guaranteed by the problem definition, given that the system is only extended and the system is assumed to fulfill every load scenario with intact initial pumps. In the final fifth, existential stage, it is checked whether the remaining system, i.e. the system without the broken pump selected in the previous stage, is able to fulfill the load scenario selected in the second stage. While considering the breakdown of a single pump here, the model can be modified to handle the simultaneous breakdown of multiple pumps, similar to the concept of K-resilience examined in Altherr et al. (2019). However, according to the requirement defined in DIN 1988-500, the operation of the system must only be ensured if one single pump breaks down.

Table 6.1 displays the sets, parameters and variables with the associated stage used for the QMIP. If applicable, the dimension of an entry is specified using the base quantities mass (M), length (L), currency (C) and time (T). The notation differs from the previous model because it is not a direct extension of the optimization model but an approach to be used separately in a subsequent planning phase.

Table 6.1: Variables, sets and parameters of the QMIP optimization model

Stage	Symbol	Range	Dimension	Description
1	b_p	$\{0,1\}$		Purchase indicator for pump $p \in A$
	a_p^i	$\{0,1\}$		Activation indicator for pump $p \in P$ in scenario $i \in S$
	c^i	$\mathbb{R}_0^+$	C	Operational costs in scenario $i \in S$
2	s	S		Scenario selection
3	σ^i	$\{0,1\}$		Indicator whether scenario $i \in S$ is selected
	x_p	$\{0,1\}$		Indicator whether pump $p \in P$ is used
	$\dot{v}_p$	$\mathbb{R}_0^+$	L^3T^{-1}	Volume flow rate through pump $p \in P$
	Δh_p	$\mathbb{R}_0^+$	$ML^{-1}T^{-2}$	Pressure increase caused by pump $p \in P$
	ρ_p	$\mathbb{R}_0^+$	ML^2T^{-3}	Power consumption of pump $p \in P$
	n_p	$[0,1]$		Relative rotational speed of pump $p \in P$
4	d	I		Selection of damaged pump
5	δ_p	$\{0,1\}$		Indicator whether initial pump $p \in I$ is damaged
	x_p^B	$\{0,1\}$		Indicator whether pump $p \in P$ is used in case of disturbance
	$\dot{v}_p^B$	$\mathbb{R}_0^+$	L^3T^{-1}	Volume flow rate through pump $p \in P$ in case of disturbance
	Δh_p^B	$\mathbb{R}_0^+$	$ML^{-1}T^{-2}$	Pressure increase caused by pump $p \in P$ in case of disturbance
	S			Set of scenarios
	I			Set of initial pumps
	A			Set of additional pumps
	P	$A \cup I$		Set of all available pumps
	C_p^{buy}	$\mathbb{R}_0^+$	C	Purchase costs of pump $p \in A$
	$\dot{V}^i$	$\mathbb{R}_0^+$	L^3T^{-1}	Required volume flow rate in scenario $i \in S$
	ΔH^i	$\mathbb{R}_0^+$	$ML^{-1}T^{-2}$	Required pressure increase in scenario $i \in S$
	OLT	$\mathbb{R}_0^+$	T	Projected operational lifetime of the system
	F^i	$[0,1]$		Projected relative occurrence of scenario $i \in S$ during the operational lifetime OLT
	C^{kWh}	$\mathbb{R}_0^+$	$M^{-1}L^{-2}CT^2$	Operation costs per kilowatt hour of electricity
	ΔH^{max}	$\mathbb{R}_0^+$	$ML^{-1}T^{-2}$	Upper bound on the pressure increase
	$\dot{V}^{max}$	$\mathbb{R}_0^+$	L^3T^{-1}	Upper bound on the volume flow rate
	M	$\mathbb{R}_0^+$		Sufficiently large constant

As mentioned above, the aim is to minimize the sum of the weighted energy costs during operation with regard to the expected load scenarios and the investment for the additional stand-by pumps, see Objective (6.1). An additional pump can only be used if it is purchased, see Constraints (6.2) and (6.3). This includes both the "regular operation" of the third stage as well as the "breakdown operation" of the fifth stage and is also used to link the first stage with the third and fifth stage. In combination, Constraints (6.4) to (6.6) ensure that each pump that is part of the system has to be operational in at least one of the load scenarios if "regular operation" is considered. All pumps from the set of initial pumps are automatically part of the system, see Constraint (6.5), while the pumps from the set of additional pumps are only part of the system if they are purchased, see Constraint (6.6). The following Constraints (6.7) to (6.10) have no technical interpretation and only serve to transform the universal integer variable of the selected scenario and the selected broken pump to the corresponding existential binary variables. Since the broken pump cannot be used during the "breakdown operation", this is prohibited by Constraint (6.11). The power consumption and pressure increase caused by a pump within its respective physical domain defined by its associated characteristic map depends on the volume flow through the pump as well as its rotational speed. This nonlinear relationship, which can be modeled using the techniques presented in Section 2.2.3, is outlined by Constraints (6.12) and (6.13). In case of the "breakdown operation", this can be modeled more easily since, as described above, the associated energy costs and therefore the power consumption are neglected due to short repair times. Hence, it is sufficient to linearize the boundaries of a pump's characteristic map and to check whether the necessary pressure increase at a certain volume flow lies within these limits, see Constraint (6.14). An explicit consideration of the rotational speed is not necessary in this case. In addition, it has to be ensured that the pressure increase caused by each pump used in the selected scenario as well as the sum of the volume flows of all pumps match the required pressure increase and volume flow exactly for both the "regular operation" and "breakdown operation", see Constraints (6.15) to (6.18). For the parallel connection of multiple pumps, the total volume flow rate is the sum of the individual volume flows and each pump generates the same common pressure increase. Furthermore, the nonlinearity in Constraints (6.15) and (6.16) can be resolved straightforwardly by using a big-M formulation. In this context, Constraints (6.19) to (6.22) establish general bounds on the volume flow and pressure increase for both operation modes and are used in order to force the volume flow and pressure increase of a pump to vanish if this pump is not used in the selected scenario. Finally, Constraint (6.23) transforms the power consumption associated with the "regular operation" for the selected scenario to energy costs.

$$\min \sum_{i \in S} F^i \cdot c^i + \sum_{p \in A} C_p^{buy} \cdot b_p \tag{6.1}$$

$$x_p \leq b_p \qquad \forall p \in A \tag{6.2}$$

$$x_p^B \leq b_p \qquad \forall p \in A \tag{6.3}$$

$$a_p^i - x_p + \sigma^i \leq 1 \qquad \forall p \in P,\ i \in S \tag{6.4}$$

$$\sum_{i \in S} a_p^i \geq 1 \qquad \forall p \in I \tag{6.5}$$

$$\sum_{i \in S} a_p^i \geq b_p \qquad \forall p \in A \tag{6.6}$$

$$\sum_{i \in S} \sigma^i = 1 \tag{6.7}$$

$$\sum_{i \in S} i \cdot \sigma^i = s \tag{6.8}$$

$$\sum_{p \in I} \delta_p = 1 \tag{6.9}$$

$$\sum_{p \in I} p \cdot \delta_p = d \tag{6.10}$$

$$x_p^B + \delta_p \leq 1 \qquad \forall p \in I \tag{6.11}$$

$$\rho_p = \mathcal{P}_p(\dot{v}_p, n_p) \qquad \forall p \in P \tag{6.12}$$

$$\Delta h_p = \Delta \mathcal{H}_p(\dot{v}_p, n_p) \qquad \forall p \in P \tag{6.13}$$

$$\Delta h_p^B = \Delta \mathcal{H}_p^B(\dot{v}_p^B) \qquad \forall p \in P \tag{6.14}$$

$$\Delta h_p = x_p \cdot \sum_{i \in S} \Delta H^i \cdot \sigma^i \qquad \forall p \in P \tag{6.15}$$

$$\Delta h_p^B = x_p^B \cdot \sum_{i \in S} \Delta H^i \cdot \sigma^i \qquad \forall p \in P \tag{6.16}$$

$$\sum_{p \in P} \dot{v}_p = \sum_{i \in S} \dot{V}^i \cdot \sigma^i \tag{6.17}$$

$$\sum_{p \in P} \dot{v}_p^B = \sum_{i \in S} \dot{V}^i \cdot \sigma^i \tag{6.18}$$

$$\dot{v}_p \leq \dot{V}^{max} \cdot x_p \qquad \forall p \in P \tag{6.19}$$

$$\Delta h_p \leq \Delta H^{max} \cdot x_p \qquad \forall p \in P \tag{6.20}$$

$$\dot{v}_p^B \leq \dot{V}^{max} \cdot x_p^B \qquad \forall p \in P \tag{6.21}$$

$$\Delta h_p^B \leq \Delta H^{max} \cdot x_p^B \qquad \forall p \in P \tag{6.22}$$

$$M \cdot (1 - \sigma^i) + c^i \geq C^{kWh} \cdot OLT \cdot \sum_{p \in P} \rho_p \qquad \forall i \in S \tag{6.23}$$

6.3 Application Examples

In order to demonstrate the approach, two artificial examples are investigated. The corresponding load scenarios are shown in Table 6.2. Again, the pumps used for this application are based on the Wilo-Economy MHIE model series already presented in Chapter 5. However, the single pump in group 16xx is neglected hereinafter due to its superiority compared to the other pumps in the considered examples. A suitable number of data points was extracted from the pumps' datasheets in order to approximate the characteristic maps.

Both generated QMIP instances are solved using the framework provided by the QMIP solver Yasol[1]. Because the solver itself can only deal with continuous variables in the final stage, the option of creating and solving the corresponding DEP is used. The runtimes of the examined examples were in the range of seconds. Hence, this subject is not deepened any further.

Table 6.2: Summary of the load scenarios for the considered test instances

	Example 1			Example 2		
	F	$\dot{V}$ [m^3/h]	ΔH [m]	F	$\dot{V}$ [m^3/h]	ΔH [m]
Scenario 1	0.1	25	20	0.25	20	30
Scenario 2	0.3	20	30	0.25	12	50
Scenario 3	0.4	15	40	0.25	10	20
Scenario 4	0.2	10	60	0.25	5	15

6.3.1 Optimized Initial Design

As a first example, a system that is already optimized regarding the sum of investment and energy costs over a predefined set of load scenarios for the non-resilient case is examined. The considered load scenarios are shown on the left-hand side of Table 6.2. This system consists of one pump each of the types 206, 403, 406 and 803 connected in parallel and has initial operational costs of €75 288.88 assuming a lifetime of ten years. In order to transform this given booster station, which can only fulfill the considered load scenarios if all pumps are fully functional, into a more resilient system, the presented optimization model is applied. The set of additional pumps A that can be added contains each pump of the Wilo-Economy

[1] http://q-mip.org (accessed May 07, 2021)

MHIE series exactly once. According to the solution of the QMIP, it is optimal to add the additional pump 205 with purchase costs of €1 805 to the system. This might seem somewhat surprising at first glance given that, even though the system was optimized for the non-resilient case, none of the already installed pumps is doubled and instead a new type is added to the network in order to compensate for the breakdown of one of the initial pumps. However, a detailed examination confirmed that this is indeed the optimal solution. This shows that even for an optimized system, finding a more resilient configuration is a non-trivial task. Compared to the original system, the selected additional pump is used in the first scenario, resulting in an increase in lifetime operational costs of only €3.52 compared to the non-resilient case. Hence, the minimal additional costs to enhance the resilience of the initial booster station are €1 808.52.

6.3.2 Multiple Identical Pumps

As a second example, the case of an initial system with multiple identical pumps connected in parallel, which follows the conventional design approach, is considered. The obvious way to achieve the addressed sense of resilience for this system is to add another pump of the same type to the network. However, cheaper configurations may exist. For this example, a system with three pumps of the 406-type is studied. The corresponding load scenarios can be found on the right-hand side of Table 6.2 and the system is projected to be operational for five years. As in the previous example, it is intended to transform the not yet resilient system into a more resilient one by adding pumps of the Wilo-Economy MHIE, whereby each pump may only be installed once. The solution of the associated QMIP recommends the purchase of the not yet existing pump type 403 as an additional pump to be included in the network. Following this recommendation, the operational costs decrease in the second, third and fourth load scenario compared to the initial system. This is due to the fact that the initial system was not optimal itself for the given load scenarios—a circumstance occurring frequently as systems are often designed to cover a broad range of conditions for various applications. In terms of the financial impact of this investment decision, €2 243.30 can be saved over the five years compared to the common approach of adding a fourth pump of the 406-type in order to increase resilience. These savings result from two different reasons: Firstly, selecting the 403-pump with a lower initial investment and secondly, being able to operate more efficiently in the individual load scenarios as a better system operating point can be reached with the addition of a 403-type pump.

7 Modeling of Thermofluid Systems

Thermofluid systems can be regarded as fluid systems with superimposed heat transfer (Pöttgen et al., 2016). Hence, thermofluid systems fulfill two subtasks, the distribution as well as heating and cooling of fluids. For modeling these systems several possibilities with different focal points are presented in literature. However, in the following, the focus is on two essential aspects: maintaining a simple representation and the compatibility with the fluid model from Chapter 4. The optimization model for fluid systems is therefore extended by the introduction of additional constraints dealing with heating and cooling. This involves introducing the physical quantity temperature T and the associated energy that is carried by the flow of fluid. As with pure fluid systems, the scope of this work is limited to incompressible liquids. Given the assumptions discussed in Section 2.1.3 for an incompressible liquid, it is convenient to use the thermal energy flow rate $\dot{U}$ in the model to account for the thermal energy that is carried by the mass flow of fluid. It should be noted that the subscript t used in Section 2.1.3, which denotes the thermal component of the internal energy, is omitted for better readability. Furthermore, in the scope considered, the thermal energy can be treated as equivalent to the internal energy since the other components of the internal energy are not relevant for the consideration. Accordingly, the thermal energy flow rate can be defined as $\dot{U} = \dot{m} \cdot u$, with mass flow rate $\dot{m}$ and specific thermal energy u. The mass flow rate can also be expressed by $\dot{m} = \dot{V} \cdot \rho$, with volume flow rate $\dot{V}$ and density ρ, which is assumed to be constant in the following. In order to overcome the difficulty related to using absolute values, the thermal energy flow rate $\dot{U}$ can be expressed relative to a certain reference state, i.e. $u_0 = u(T_0)$, since only differences and not absolute values are relevant (Doran, 2011). By assuming the liquid to be a calorically perfect material, i.e. constant specific heat c, in the considered temperature range, it is possible to state that

$$u = u_0 + c \cdot \big(T - T_0\big) \tag{7.1}$$

and hence

$$\dot{U} = \dot{m} \cdot \big(u_0 + c \cdot \big(T - T_0\big)\big). \tag{7.2}$$

Since this only concerns the consideration of temperature differences, it is not relevant whether the temperatures are specified in Kelvin or degree Celsius. For convenience, the reference state for the model is chosen to be $u_0 = u(0°\mathrm{C}) = 0$ J/kg. This has two benefits: Firstly, $(T - T_0)$ directly corresponds to the actual temperature measured in degree Celsius and secondly, $\dot{U} \geq 0$ W for the considered temperature range.

If a heating or cooling component is present, it is then assumed that there is a change in the thermal energy flow rate $\Delta\dot{U}$ between the inlet and outlet of the component according to the heat flow $\dot{Q}$ transferred to the fluid by this component or vice versa, i.e. $\Delta\dot{U} = \dot{Q}$. For other components, it is assumed that this change in thermal energy flow rate is negligible.

In order to be able to establish this extension, two further aspects have to be explicitly addressed. These occur when several flows are merged or split up at junctions (Elmqvist et al., 2003). The former occurs for incoming and the latter for outgoing flows. Even though in general merging and splitting up flows is also common in pure fluid systems, the distinction and the reason for the increased complexity is that the considered flows may have different temperatures and that these temperatures are directly related to the other system variables.

In the case that several flows enter a junction, perfect mixing is assumed. Hence, with N flows entering a junction k, the specific thermal energy at the junction u_k and the related temperature T_k are given by

$$u_k = \frac{\sum\limits_{i \in N} \dot{m}_i \cdot u_i}{\sum\limits_{i \in N} \dot{m}_i} = \frac{\sum\limits_{i \in N} \dot{U}_i}{\sum\limits_{i \in N} \dot{m}_i} \sim T_k \tag{7.3}$$

as indicated in Elmqvist et al. (2003). With $\dot{m} = \dot{V} \cdot \rho$, the temperature resulting from all incoming flows can therefore be expressed as a function of the sum of volume flows and associated thermal energy flow rates entering a certain junction. In the case of only one flow entering a junction, the mixing temperature is the temperature of the incoming flow. Equivalently, the resulting mixing temperature T_k at the junction can be determined by Richmann's calorimetric mixing formula (see e.g. Tillmann and Bohn, 2015) as follows:

$$T_k = \frac{\sum\limits_{i \in N} \dot{m}_i \cdot T_i \cdot c}{\sum\limits_{i \in N} \dot{m}_i \cdot c} \tag{7.4}$$

In the case of outgoing flows, the specific thermal energies for all flows leaving a junction are the same as the specific thermal energy at the junction (Elmqvist et al., 2003). Hence, for M flows leaving a junction k with specific thermal energy u_k, the specific thermal energies of each individual flow u_j and the related temperatures T_j become:

$$u_k = u_j = \frac{\dot{U}_j}{\dot{m}_j} \sim T_k = T_j \qquad \forall\, j \in M \tag{7.5}$$

Accordingly, all flows leaving the same junction have the same temperature. With $\dot{m} = \dot{V} \cdot \rho$, the temperature at the junction therefore determines the (constant) ratio between the volume flow rate and the associated thermal energy flow rate for each individual flow leaving this junction.

7.1 Components of Thermofluid Systems

There is a wide variety of different components used in thermofluid systems depending on the respective field of application. However, two ideal sources of thermal energy can be distinguished: ideal heat sources and ideal temperature sources (Pöttgen et al., 2016). An ideal heat source is able to maintain a constant, predefined heat flow to the fluid. An ideal temperature source in contrast can maintain a predefined temperature at its outlet. Here, all heating and cooling components are associated with one of the two ideal sources of thermal energy. In the considered scope, it is possible to define more detailed representations of different components, even if they differ in their respective behavior or have special features, due to the decoupling of the general system behavior and component characteristics. However, the base characteristics of a component are still determined by its respective affiliation to one of the two ideal sources of thermal energy. In the following, two examples are given: boilers used in heating systems as representatives of a component type that can be modeled as a heat source and chillers as representatives of a component type that can be modeled as a temperature source. Besides this general classification, there are also other individual distinguishing features that have to be accounted for in the modeling process such as the ability of components to operate only at a fixed operating point or whether the operating point can be changed step-wise or continuously.

7.1.1 Boilers

In the considered scope, boilers are components that either burn some kind of fuel, e.g. gas, oil or biomass, or use electricity to provide a heat flow to the fluid. For the description of (tankless) boilers used here, the model for a biomass boiler with discrete operating points used in Pöttgen et al. (2016) is adapted. Nevertheless, it is straightforward to transfer this approach to other technologies.

The considered boiler has multiple discrete operating points. For example, the boiler may be operated at full load or at half load. At full load, the boiler provides a certain constant maximum heat flow to the passing fluid. At half load, this heat flow is half the value of that maximum heat flow. This is assumed to be independent of the volume flow through the boiler as well as the temperature of the incoming fluid. Hence, for a boiler with $|L|$ discrete, equally spaced operating points, its operating level can be expressed by an integer variable l with $l \in \{0, \dots, |L|\}$. With this, the heat flow provided by the boiler can be expressed as

$$\dot{Q} = \dot{Q}^{max} \cdot \frac{l}{|L|}, \tag{7.6}$$

where $\dot{Q}^{max}$ is the heat flow at full load. If l takes the value zero, there is no heat transfer, which corresponds to a switched off state. The switching between these states is assumed to happen instantaneously. Possible transition times are deliberately not considered because it is assumed that they are negligible compared to the duration of the boiler's stationary operation. The corresponding energy consumption per unit time is estimated on the basis of the heat flow provided to the fluid, taking into account the respective energy source and assuming a component-specific, constant efficiency. For a more detailed approach including part-load dependent efficiency, see e.g. Voll (2013) and Voll et al. (2013).

7.1.2 Chillers

In general, many different chiller types exist. However, there are two main classes of chillers: vapor absorption and vapor compression chillers. Compression chillers use mechanical compressors driven by electric motors, while absorption chillers are driven by a heat source. In the following, the focus is on vapor compression chillers, for which different compressor types exist. The available compressor technologies include scroll and reciprocating compressors for small cooling capacities, screw compressors for medium capacities, and centrifugal compressors for large cooling capacities. A further distinction can be made with regard to the heat sink used for rejecting heat. Compression chillers can be either water-cooled or air-cooled. While water-cooled chillers reject heat using the cooling water of a cooling tower, air-cooled chillers reject heat directly to the ambient air. As shown in Figure 7.1, the cooling is realized by a circular process consisting of four subprocesses (Baglione, 2011): At the start of this cycle, the internal refrigerant is a saturated vapor (1). In the compressor, the pressure of the refrigerant is increased resulting in a superheated vapor (2). Afterwards, the superheated vapor enters the condenser and rejects heat to the ambient air or water of a cooling tower and liquefies again, becoming a saturated liquid (3). Passing the expansion valve, the pressure of the refrigerant is abruptly reduced, resulting in a liquid-vapor mixture (4). Finally, in the evaporator, heat of the cooling medium returning from the heat source is absorbed, completing the cycle, which is then repeated.

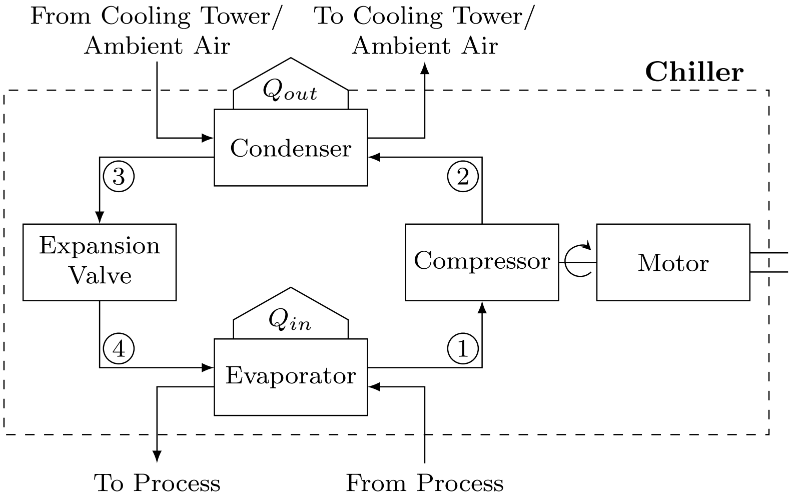


Figure 7.1: Illustration of the working principle of compression chillers (based on Baglione, 2011)

In order to describe chillers and their complex behavior, a suitable model has to be determined. For representing the operation of chillers, a large number of (semi-)empirical models with varying level of detail are discussed in literature. Examples include models based on linear regression, bi-quadratic regression, multivariate polynomial regression as well as different variations of the DOE-2 and the

Gordon-Ng model (Lee et al., 2012). However, most studies focusing on the more detailed chiller models, e.g. the DOE-2 (see e.g. Zhang et al., 2017) or the Gordon-Ng model (see e.g. Powell et al., 2013), examine the operation of a predefined system. Due to the comparatively high degree of abstraction of the design level and the associated uncertainties regarding the actual operation conditions as well as the additional difficulty resulting from the two-stage character of the problem under consideration, a deliberately simpler approach based on the model discussed in Deng et al. (2015) is chosen instead. While this approach neglects certain influences and interactions, it was shown to be suitable to sufficiently represent the behavior of real-world applications (see Deng et al., 2015). Furthermore, unlike other simpler models (see e.g. Voll et al., 2013), quality levels, i.e. the consideration of temperature, can be addressed rather than relying solely on energy balances. Nevertheless, since the component and the system behavior are deliberately separated in the presented MILP and the focus is on the interaction of components rather than their detailed internal behavior, it is possible to use other models to predict a chiller's operation as long as the relevant system in- and outputs can be incorporated.

In the general context of refrigerators and heat pumps, the so-called Coefficient of Performance (COP) is an often used and important relationship for analyzing their performance. The COP acts as a metric used to quantify the efficiency and expresses the ratio between the available useful cooling or heating $\dot{Q}$ and required (e.g. electrical) power input P at a given operating point. It should be mentioned that a distinction is often made where the COP is used for heating performance, while the so-called Energy Efficiency Ratio (EER) is used for cooling performance. However, due to the analogous determination of the EER and the widespread use of the COP in common parlance, the term "COP" is used in the following. For chillers, the COP can be stated as

$$COP = \frac{\dot{Q}}{P}. \tag{7.7}$$

As can be seen, a higher COP yields a higher efficiency. The variable $\dot{Q}$ is called "cooling capacity" and represents a chiller's ability to provide cooling. It can be expressed as

$$\dot{Q} = \dot{V} \cdot \rho \cdot c \cdot (T_{chwr} - T_{chws}), \tag{7.8}$$

with volume flow rate $\dot{V}$, (constant) density ρ, (constant) specific heat capacity c, chilled water return temperature T_{chwr} and chilled water supply temperature T_{chws}.

However, according to the chiller model presented in Deng et al. (2015), it is assumed that chillers are only operated in so-called on/off modes. This means that if a chiller is turned on, it supplies chilled water at a certain constant design flow rate $\dot{V}^N$ and constant supply temperature T^N_{chws} or it is turned off otherwise. The respective mode of a chiller can be represented by a binary indicator a with $a = 1$ if a chiller is turned on and $a = 0$ if a chiller is turned off. Hence, two cases can be distinguished:

$$\dot{Q} = \begin{cases} \dot{V}^N \cdot \rho \cdot c \cdot \left(T_{chwr} - T^N_{chws}\right), & \text{if } a = 1 \\ 0, & \text{otherwise.} \end{cases} \tag{7.9}$$

The cooling provided varies with respect to the return temperature, which in turn results from the operating point of the system, while the supply temperature is kept constant.

Since the volume flow rate is constant and the cooling capacity therefore only depends on the temperature difference, the power consumption P is assumed to be a function of the difference in temperature only. Hence, the function for determining the electrical power consumption takes the following form

$$P = \begin{cases} \alpha \cdot \left(T_{chwr} - T^N_{chws}\right) + \beta, & \text{if } a = 1 \\ 0, & \text{otherwise} \end{cases} \tag{7.10}$$

with linear regression coefficients α and β. This function can be obtained by linear regression of either measured or manufacturer data (Deng et al., 2015). It is essential to pay attention to the data range for which the relationship is obtained. Outside of this range and using a different supply temperature or volume flow, the performance is extrapolated and may behave unexpectedly.

Figure 7.2 shows an exemplary plot of a chiller's COP with varying temperature differences ($T_{chwr} - T_{chws}$) in the calibrated range $[5, 15]$°C with T_{chws} being held constant. In general, there are other influencing variables, e.g. condenser water temperature or ambient temperature, which are neglected in order to reduce complexity.

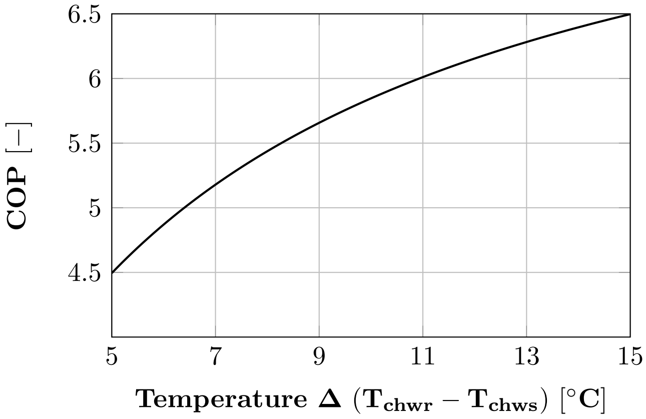


Figure 7.2: Exemplary illustration of a chiller's COP as a function of the difference between the return and supply temperatures ($T_{chwr} - T_{chws}$) in the range $[5, 15]$°C and constant T_{chws} (based on Deng et al., 2015)

7.2 Extension of the Optimization Model for Thermofluid Systems

As described above, the optimization model for fluid systems from Chapter 4 has to be extended by additional constraints dealing with heating and cooling. This involves introducing the physical quantities temperature and thermal energy flow rate and their interactions in the model as well as taking care of the additional component groups necessary for heating and cooling. Since this is an extension of fluid systems, only the additional constraints are discussed here and the previously presented constraints of Chapter 4 still apply. Hence, the full MILP results from joining both parts. As for fluid systems, the representation is split into two parts, the system behavior and the component characteristics. Again, the symbols used within the model should not be confused with the symbols used in other contexts, as they may differ in some cases. Therefore, all variables and parameters used are shown in Table 7.1. If applicable, the dimension of an entry is specified using the base quantities mass (M), length (L), currency (C), time (T) and temperature (K). The entries only applying to the component-specific parts are separated by dashed lines.

Table 7.1: Variables, sets and parameters of the optimization model for thermofluid systems

Symbol	Range	Dimension	Description
$b_{i,j}$	$\{0,1\}$		Purchase indicator of edge $(i,j) \in E$
$a^s_{i,j}$	$\{0,1\}$		Activation indicator of edge $(i,j) \in E$ in scenario $s \in S$
$p^s_{i,j}$	$\mathbb{R}^+_0$	ML^2T^{-3}	Power consumption caused by edge $(i,j) \in E$ in scenario $s \in S$
$\dot{v}^s_{i,j}$	$\mathbb{R}^+_0$	L^3T^{-1}	Volume flow along edge $(i,j) \in E$ in scenario $s \in S$
h^s_k	$\mathbb{R}^+_0$	$ML^{-1}T^{-2}$	Pressure at vertex $k \in V$ in scenario $s \in S$
$\Delta h^s_{i,j}$	$\mathbb{R}$	$ML^{-1}T^{-2}$	Change in pressure caused by edge $(i,j) \in E$ in scenario $s \in S$
$\dot{u}^{in\ s}_{i,j}$	$\mathbb{R}^+_0$	ML^2T^{-3}	Thermal energy flow rate directly after the start of edge $(i,j) \in E$ in scenario $s \in S$
$\dot{u}^{out\ s}_{i,j}$	$\mathbb{R}^+_0$	ML^2T^{-3}	Thermal energy flow rate directly before the end of edge $(i,j) \in E$ in scenario $s \in S$
t^s_k	$\mathbb{R}^+_0$	K	Temperature at vertex $k \in V$ in scenario $s \in S$
$\Delta\dot{u}^s_{i,j}$	$\mathbb{R}$	ML^2T^{-3}	Change in thermal energy flow rate caused by edge $(i,j) \in E \backslash TS(E)$ in scenario $s \in S$
$t^s_{i,j}$	$\mathbb{R}^+_0$	K	Outlet temperature caused by temperature source edge $(i,j) \in TS(E)$ in scenario $s \in S$
$n^s_{i,j}$	$[0,1]$		Relative rotational speed for pump edge $(i,j) \in Pu(E)$ in scenario $s \in S$
$l^s_{i,j}$	$\mathbb{N}_0$		Operating level of boiler edge $(i,j) \in Bo(E)$ in scenario $s \in S$
S			Set of scenarios
E			Set of edges
V			Set of vertices
$S_G(V)$	$\subseteq V$		Subset of source vertices
$T_G(V)$	$\subseteq V$		Subset of sink vertices
$TS(E)$	$\subseteq E$		Subset of (ideal) temperature sources
$Pu(E)$	$\subseteq E$		Subset of pump edges
$Pi(E)$	$\subseteq E$		Subset of pipe edges
$GV(E)$	$\subseteq E$		Subset of gate valve edges
$Bo(E)$	$\subseteq E$		Subset of boiler edges
$Ch(E)$	$\subseteq TS(E)$		Subset of chiller edges
$C^{buy}_{i,j}$	$\mathbb{R}^+_0$	C	Purchase costs of edge $(i,j) \in E$
$C^{kWh}_{i,j}$	$\mathbb{R}^+_0$	$M^{-1}L^{-2}CT^2$	Energy costs per kilowatt hour for edge $(i,j) \in E$
OLT	$\mathbb{R}^+_0$	T	Expected operational lifetime of the system

Continued on next page

Table 7.1: Continued from previous page

Symbol	Range	Dimension	Description
F^s	$[0, 1]$		Portion of time for which scenario $s \in S$ is expected to occur
$\dot{V}^{max}$	$\mathbb{R}_0^+$	L^3T^{-1}	General upper bound on the volume flow
H^{max}	$\mathbb{R}_0^+$	$ML^{-1}T^{-2}$	General upper bound on the pressure
$\dot{V}_{in/out\,k}^{min/max\,s}$	$\mathbb{R}_0^+$	L^3T^{-1}	Lower and upper bound on the volume flow entering and leaving vertex k in scenario s
$H_k^{min/max\,s}$	$\mathbb{R}_0^+$	$ML^{-1}T^{-2}$	Lower and upper bound on the pressure at vertex $k \in V$ in scenario $s \in S$
$\dot{U}^{max}$	$\mathbb{R}_0^+$	ML^2T^{-3}	General upper bound on the thermal energy flow rate
T^{max}	$\mathbb{R}_0^+$	K	General upper bound on the temperature
$T_k^{min/max\,s}$	$\mathbb{R}_0^+$	K	Lower and upper bound on the temperature at vertex $k \in V$ in scenario $s \in S$
$\dot{V}_{i,j}^{min/max}$	$\mathbb{R}_0^+$	L^3T^{-1}	Lower and upper bound on the volume flow for edge $(i, j) \in E$
$N_{i,j}^{min/max}$	$[0, 1]$		Lower and upper bound on the rotational speed for pump edge $(i, j) \in Pu(E)$
$P_{i,j}^{max}$	$\mathbb{R}_0^+$	ML^2T^{-3}	Upper bound on the power consumption by edge $(i, j) \in E$
$L_{i,j}^{max}$	$\mathbb{N}$		Upper bound on the operating level of boiler edge $(i, j) \in Bo(E)$
$\Delta\dot{U}_{i,j}^{full}$	$\mathbb{R}_0^+$	ML^2T^{-3}	Change in thermal energy flow rate caused by boiler edge $(i, j) \in Bo(E)$ at full load
$T_{i,j}^{out}$	$\mathbb{R}_0^+$	K	Outlet temperature of chiller edge $(i, j) \in Ch(E)$
$T_{i,j}^{in\,min/max}$	$\mathbb{R}_0^+$	K	Lower and upper bound on the inlet temperature of chiller edge $(i, j) \in Ch(E)$
$\alpha_{i,j}$, $\beta_{i,j}$	$\mathbb{R}$		Regression coefficients for determining the power consumption by chiller edge $(i, j) \in Ch(E)$
$\mathcal{T}(*)$	$\mathbb{R}_0^+$	K	Representation of the nonlinear temperature relationship
$\Delta\mathcal{H}_{i,j}(*)$	$\mathbb{R}$	$ML^{-1}T^{-2}$	Representation of the nonlinear relationship for the change in pressure caused by edge $(i, j) \in E$
$\mathcal{P}_{i,j}(*)$	$\mathbb{R}_0^+$	ML^2T^{-3}	Representation of the nonlinear relationship for the power consumption caused by edge $(i, j) \in Pu(E)$

7.2.1 Objective Function

The objective function for thermofluid systems remains almost unchanged compared to fluid systems. The only difference is that the cost parameter for determining energy costs is now also dependent on the respective component represented by the indices $(i,j) \in E$, see Objective (7.11). This is necessary in order to be able to include not only components powered by electrical energy but also other energy sources such as gas or biomass, which are used frequently in thermofluid systems.

$$\min \quad \sum_{(i,j)\in E} \Big(C^{buy}_{i,j} \cdot b_{i,j} \Big) + OLT \cdot \sum_{s \in S} \Big(F^s \cdot \Big(\sum_{(i,j)\in E} C^{kWh}_{i,j} \cdot p^s_{i,j} \Big) \Big) \tag{7.11}$$

7.2.2 General System Constraints

In the following, Constraints (7.12a) to (7.12i), already present in the fluid system model, are grayed out but displayed for the sake of completeness. See Chapter 4 for the associated explanations. Furthermore, it should be mentioned that the thermal energy flow rate can change along edges due to heating and cooling, i.e. heat transfer, provided to the system by the components that are represented by these edges. Hence, two variables rather than one are required to model it. The variable $\dot{u}^{in\,s}_{i,j}$ represents the thermal energy flow rate directly behind the start vertex of an edge, corresponding to the thermal energy flow rate into the component. Accordingly, $\dot{u}^{out\,s}_{i,j}$ represents the thermal energy flow rate directly before the end vertex of an edge, corresponding to the thermal energy flow rate out of the component.

If an edge is not active, the thermal energy flow rate along this edge has to vanish, see Constraints (7.12j) and (7.12k). Constraint (7.12l) establishes a general upper bound on the temperature for all vertices. Due to the law of energy conservation, the net thermal energy flow rate has to be zero for all vertices, except for the source and sink vertices, see Constraint (7.12m). Hence, it must be ensured that the total thermal energy flow rate into a vertex is equal to the total thermal energy flow rate out of that vertex. It is possible that there are exceptions other than source and sink vertices, such as storage vertices, which are able to store a certain amount of thermal energy. For these cases, it is necessary to introduce a variable $\Delta\dot{u}^s_k$ representing the difference between both rates and modify Constraint (7.12m) accordingly. However, for the setting considered here, it is not necessary. The temperature at a vertex can be determined by the total volume flow and associated total thermal energy flow rate into that vertex, see Constraint (7.12n). In this context, it is assumed that the density ρ and specific heat capacity c are constant. Since there is no flow into a source vertex, this vertex is excluded. The nonlinear relationship for the temperature is modeled as a piecewise linear approximation using the linearization techniques presented in Section 2.2.3. In contrast to the pure fluid system model, this implies that nonlinearities are not only present for the component behavior but also for the general system behavior itself. Furthermore, each flow leaving the same vertex by an active edge must have the same temperature, which is the temperature at that vertex, see Constraints (7.12o) and (7.12p). However, if there is only one flow leaving a vertex, this condition is automatically fulfilled. If an edge is active, the thermal energy flow rate along this edge changes according to the change in thermal energy flow rate caused by the associated component, see Constraints (7.12q) and (7.12r). Equality could be requested here instead since there is no thermal energy flow rate along an inactive edge due to the Constraints (7.12j) and (7.12k). However, for illustration purposes, this form is chosen in order to clarify that these constraints deal with the change of the thermal energy flow rate for active edges only. Furthermore, there is an exception as two sources of thermal energy are differentiated: ideal heat sources and ideal temperature sources. As mentioned above, an ideal heat source maintains a constant heat flow to the system and an ideal temperature source maintains a constant temperature at its outlet. In the case of ideal temperature sources denoted by $TS(E)$, Constraints (7.12q) and (7.12r) are substituted by Constraints (7.12s) and (7.12t). Hence, a constant temperature is assigned to the end vertex of an edge if this edge is active, see Constraints (7.12s) and (7.12t). Finally, Constraint (7.12u) enables the setting of target ranges for the temperature at certain points in the system or can be used to limit its values. By equating the respective lower and upper bound, target values instead of ranges can be specified. This is primarily used to specify the temperature for source and sink vertices.

$$a^s_{i,j} \leq b_{i,j} \qquad \forall\, s \in S,\ (i,j) \in E \tag{7.12a}$$

$$\dot{v}^s_{i,j} \leq \dot{V}^{max} \cdot a^s_{i,j} \qquad \forall\, s \in S,\ (i,j) \in E \tag{7.12b}$$

$$h^s_k \leq H^{max} \qquad \forall\, s \in S,\ k \in V \tag{7.12c}$$

$$\sum_{(i,k)\in E} \dot{v}^s_{i,k} - \sum_{(k,j)\in E} \dot{v}^s_{k,j} = 0 \qquad \forall\, s \in S,\ k \in V \backslash (S_G(V) \cup T_G(V)) \tag{7.12d}$$

$$h^s_j - h^s_i \leq \Delta h^s_{i,j} + H^{max} \cdot (1 - a^s_{i,j}) \qquad \forall\, s \in S,\ (i,j) \in E \tag{7.12e}$$

$$h^s_j - h^s_i \geq \Delta h^s_{i,j} - H^{max} \cdot (1 - a^s_{i,j}) \qquad \forall\, s \in S,\ (i,j) \in E \tag{7.12f}$$

$$\dot{V}_{out\,k}^{min\,s} \leq \sum_{(k,j)\in E} \dot{v}_{k,j}^{s} \leq \dot{V}_{out\,k}^{max\,s} \qquad \forall\, s \in S,\ k \in V \tag{7.12g}$$

$$\dot{V}_{in\,k}^{min\,s} \leq \sum_{(i,k)\in E} \dot{v}_{i,k}^{s} \leq \dot{V}_{in\,k}^{max\,s} \qquad \forall\, s \in S,\ k \in V \tag{7.12h}$$

$$H_{k}^{min\,s} \leq h_{k}^{s} \leq H_{k}^{max\,s} \qquad \forall\, s \in S,\ k \in V \tag{7.12i}$$

$$\dot{u}_{i,j}^{in\,s} \leq \dot{U}^{max} \cdot a_{i,j}^{s} \qquad \forall\, s \in S,\ (i,j) \in E \tag{7.12j}$$

$$\dot{u}_{i,j}^{out\,s} \leq \dot{U}^{max} \cdot a_{i,j}^{s} \qquad \forall\, s \in S,\ (i,j) \in E \tag{7.12k}$$

$$t_{k}^{s} \leq T^{max} \qquad \forall\, s \in S,\ k \in V \tag{7.12l}$$

$$\sum_{(i,k)\in E} \dot{u}_{i,k}^{out\,s} - \sum_{(k,j)\in E} \dot{u}_{k,j}^{in\,s} = 0 \qquad \forall\, s \in S,\ k \in V \backslash (S_G(V) \cup T_G(V)) \tag{7.12m}$$

$$t_{k}^{s} = \mathcal{T}\Big(\sum_{(i,k)\in E} \dot{v}_{i,k}^{s}, \sum_{(i,k)\in E} \dot{u}_{i,k}^{out\,s} \Big) \qquad \forall\, s \in S,\ k \in V \backslash S_G(V) \tag{7.12n}$$

$$\mathcal{T}(\dot{v}_{i,j}^{s}, \dot{u}_{i,j}^{in\,s}) \leq t_{i}^{s} + T^{max} \cdot (1 - a_{i,j}^{s}) \qquad \forall\, s \in S,\ (i,j) \in E \tag{7.12o}$$

$$\mathcal{T}(\dot{v}_{i,j}^{s}, \dot{u}_{i,j}^{in\,s}) \geq t_{i}^{s} - T^{max} \cdot (1 - a_{i,j}^{s}) \qquad \forall\, s \in S,\ (i,j) \in E \tag{7.12p}$$

$$\dot{u}_{i,j}^{out\,s} \leq \dot{u}_{i,j}^{in\,s} + \Delta\dot{u}_{i,j}^{s} + \dot{U}^{max} \cdot (1 - a_{i,j}^{s}) \qquad \forall\, s \in S,\ (i,j) \in E \backslash TS(E) \tag{7.12q}$$

$$\dot{u}_{i,j}^{out\,s} \geq \dot{u}_{i,j}^{in\,s} + \Delta\dot{u}_{i,j}^{s} - \dot{U}^{max} \cdot (1 - a_{i,j}^{s}) \qquad \forall\, s \in S,\ (i,j) \in E \backslash TS(E) \tag{7.12r}$$

$$t_{j}^{s} \leq t_{i,j}^{s} + T^{max} \cdot (1 - a_{i,j}^{s}) \qquad \forall\, s \in S,\ (i,j) \in TS(E) \tag{7.12s}$$

$$t_{j}^{s} \geq t_{i,j}^{s} - T^{max} \cdot (1 - a_{i,j}^{s}) \qquad \forall\, s \in S,\ (i,j) \in TS(E) \tag{7.12t}$$

$$T_{k}^{min\,s} \leq t_{k}^{s} \leq T_{k}^{max\,s} \qquad \forall\, s \in S,\ k \in V \tag{7.12u}$$

7.2.3 Component-Specific Constraints

In the following, the model extensions for fluid components as well as the models for the exemplary discussed heating and cooling components from Section 7.1 are presented.

Pumps, Pipes and Gate Valves

The behavior of the components already introduced in the context of pure fluid systems has to be extended in order to be applicable to the consideration of thermofluid systems. These components are pumps, pipes and gate valves, see Constraints (7.13), (7.14) and (7.15). As with the general system constraints, constraints already present in the fluid system model are grayed out but displayed for the sake of completeness. See Chapter 4 for the associated explanations.

Pumps, pipes and gate valves are in general assumed to not change the thermal energy flow rate directly. They interact with the thermal characteristics of the system only through possible (unintended) heat losses or gains to or from the environment. In the case of pumps and gate valves, this change is neglected, see Constraint (7.13i) and (7.15d). For pipes, it is a little bit more complex. However, for the considered scope, changes caused by pipes are neglected by assuming them to be well insulated, see Constraints (7.14e). If the application explicitly requires the consideration of these effects, the change in the thermal energy flow rate caused by these components $\Delta\dot{u}^s_{i,j}$ needs to be considered. This can be integrated by introducing a suitable function for determining the expected heat losses or gains with respect to the actual application.

$$
\begin{aligned}
\dot{V}^{min}_{i,j} \cdot a^s_{i,j} \leq \dot{v}^s_{i,j} \leq \dot{V}^{max}_{i,j} \cdot a^s_{i,j} \qquad & \forall s \in S,\ (i,j) \in Pu(E) && (7.13a)\\
N^{min}_{i,j} \cdot a^s_{i,j} \leq n^s_{i,j} \leq N^{max}_{i,j} \cdot a^s_{i,j} \qquad & \forall s \in S,\ (i,j) \in Pu(E) && (7.13b)\\
\Delta h^s_{i,j} \leq \Delta\mathcal{H}_{i,j}\left(\dot{v}^s_{i,j}, n^s_{i,j}\right) + \Delta H^{max}_{i,j} \cdot (1 - a^s_{i,j}) \qquad & \forall s \in S,\ (i,j) \in Pu(E) && (7.13c)\\
\Delta h^s_{i,j} \geq \Delta\mathcal{H}_{i,j}\left(\dot{v}^s_{i,j}, n^s_{i,j}\right) - \Delta H^{max}_{i,j} \cdot (1 - a^s_{i,j}) \qquad & \forall s \in S,\ (i,j) \in Pu(E) && (7.13d)\\
0 \leq \Delta h^s_{i,j} \leq \Delta H^{max}_{i,j} \cdot a^s_{i,j} \qquad & \forall s \in S,\ (i,j) \in Pu(E) && (7.13e)\\
p^s_{i,j} \leq \mathcal{P}_{i,j}\left(\dot{v}^s_{i,j}, n^s_{i,j}\right) + P^{max}_{i,j} \cdot (1 - a^s_{i,j}) \qquad & \forall s \in S,\ (i,j) \in Pu(E) && (7.13f)\\
p^s_{i,j} \geq \mathcal{P}_{i,j}\left(\dot{v}^s_{i,j}, n^s_{i,j}\right) - P^{max}_{i,j} \cdot (1 - a^s_{i,j}) \qquad & \forall s \in S,\ (i,j) \in Pu(E) && (7.13g)\\
p^s_{i,j} \leq P^{max}_{i,j} \cdot a^s_{i,j} \qquad & \forall s \in S,\ (i,j) \in Pu(E) && (7.13h)\\
\Delta\dot{u}^s_{i,j} = 0 \qquad & \forall s \in S,\ (i,j) \in Pu(E) && (7.13i)
\end{aligned}
$$

$$
\begin{aligned}
b_{i,j} = a^s_{i,j} \qquad & \forall s \in S,\ (i,j) \in Pi(E) && (7.14a)\\
\dot{V}^{min}_{i,j} \cdot a^s_{i,j} \leq \dot{v}^s_{i,j} \leq \dot{V}^{max}_{i,j} \cdot a^s_{i,j} \qquad & \forall s \in S,\ (i,j) \in Pi(E) && (7.14b)\\
\Delta h^s_{i,j} = \Delta\mathcal{H}_{i,j}\left(\dot{v}^s_{i,j}\right) \qquad & \forall s \in S,\ (i,j) \in Pi(E) && (7.14c)\\
p^s_{i,j} = 0 \qquad & \forall s \in S,\ (i,j) \in Pi(E) && (7.14d)\\
\Delta\dot{u}^s_{i,j} = 0 \qquad & \forall s \in S,\ (i,j) \in Pi(E) && (7.14e)
\end{aligned}
$$

$$
\begin{aligned}
\dot{V}^{min}_{i,j} \cdot a^s_{i,j} \leq \dot{v}^s_{i,j} \leq \dot{V}^{max}_{i,j} \cdot a^s_{i,j} \qquad & \forall s \in S,\ (i,j) \in GV(E) && (7.15a)\\
\Delta h^s_{i,j} = 0 \qquad & \forall s \in S,\ (i,j) \in GV(E) && (7.15b)\\
p^s_{i,j} = 0 \qquad & \forall s \in S,\ (i,j) \in GV(E) && (7.15c)\\
\Delta\dot{u}^s_{i,j} = 0 \qquad & \forall s \in S,\ (i,j) \in GV(E) && (7.15d)
\end{aligned}
$$

Boilers

The special type of boiler with discrete, equally spaced operating levels considered here is used as an example for modeling a heat source edge. Constraint (7.16a) couples the operating level with the associated activation indicator. Hence, it is ensured that a non-zero operating level can only occur if the component is active. Possible restrictions on the minimum and maximum volume flows for a specific component are expressed by Constraint (7.16b). The pressure loss is neglected, see Constraint (7.16c). The change in the thermal energy flow rate between its inlet and outlet is determined by the operating level, where the expression in parentheses represents the uniform difference of the heat flow transferred to the fluid per operating level step, see Constraint (7.16d). This is further used to estimate the energy consumption for the respective energy source per time unit, see Constraint (7.16e). For this, a constant, load-independent efficiency is assumed.

$$
\begin{aligned}
l_{i,j}^{s} &\leq L_{i,j}^{max} \cdot a_{i,j}^{s} && \forall s \in S,\ (i,j) \in Bo(E) && (7.16a)\\
\dot{V}_{i,j}^{min} \cdot a_{i,j}^{s} \leq \dot{v}_{i,j}^{s} &\leq \dot{V}_{i,j}^{max} \cdot a_{i,j}^{s} && \forall s \in S,\ (i,j) \in Bo(E) && (7.16b)\\
\Delta h_{i,j}^{s} &= 0 && \forall s \in S,\ (i,j) \in Bo(E) && (7.16c)\\
\Delta \dot{u}_{i,j}^{s} &= \left(\Delta \dot{U}_{i,j}^{full} / L_{i,j}^{max}\right) \cdot l_{i,j}^{s} && \forall s \in S,\ (i,j) \in Bo(E) && (7.16d)\\
p_{i,j}^{s} &= \Delta \dot{u}_{i,j}^{s} && \forall s \in S,\ (i,j) \in Bo(E) && (7.16e)
\end{aligned}
$$

Chillers

In the context of this thesis, chillers are an example for a component group modeled as a temperature source edge. As discussed above, the applied chiller model assumes a constant design flow rate through the chiller if it is active, see Constraint (7.17a). The pressure loss can be determined by manufacturer data if necessary, see Constraint (7.17b). For the considered chiller model, the outlet temperature of a chiller is set to the respective component-specific, constant design supply temperature, see Constraint (7.17c). If a chiller is active, its power consumption is estimated by the empirical function (7.10) presented in Section 7.1.2 and depends on the difference between its supply and return temperatures, see Constraints (7.17d) and (7.17e). Otherwise, the power consumption is forced to vanish by Constraint (7.17f) if the chiller is not active. Constraints (7.17g) and (7.17h) guarantee that a chiller is only active if its return temperature is in a certain range. This has two reasons. Firstly, it can be ensured that a chiller only operates in a certain efficient operating range, e.g. with high efficiency. Secondly, the empirical relationship for determining the power consumption only holds for the temperature range it is calibrated for and may yield large errors outside of that range. In this context, particular care must be taken to ensure that the minimum return temperature is strictly above the supply temperature.

$$\dot{v}^s_{i,j} = \dot{V}_{i,j} \cdot a^s_{i,j} \qquad \forall s \in S,\ (i,j) \in Ch(E) \tag{7.17a}$$

$$\Delta h^s_{i,j} = \Delta \mathcal{H}_{i,j}\big(\dot{v}^s_{i,j}\big) \qquad \forall s \in S,\ (i,j) \in Ch(E) \tag{7.17b}$$

$$t^s_{i,j} = T^{out}_{i,j} \qquad \forall s \in S,\ (i,j) \in Ch(E) \tag{7.17c}$$

$$p^s_{i,j} \leq \alpha_{i,j} \cdot (t^s_i - t^s_{i,j}) + \beta_{i,j} + P^{max}_{i,j} \cdot (1 - a^s_{i,j}) \qquad \forall s \in S,\ (i,j) \in Ch(E) \tag{7.17d}$$

$$p^s_{i,j} \geq \alpha_{i,j} \cdot (t^s_i - t^s_{i,j}) + \beta_{i,j} - P^{max}_{i,j} \cdot (1 - a^s_{i,j}) \qquad \forall s \in S,\ (i,j) \in Ch(E) \tag{7.17e}$$

$$p^s_{i,j} \leq P^{max}_{i,j} \cdot a^s_{i,j} \qquad \forall s \in S,\ (i,j) \in Ch(E) \tag{7.17f}$$

$$t^s_i \geq T^{in\,min}_{i,j} \cdot a^s_{i,j} \qquad \forall s \in S,\ (i,j) \in Ch(E) \tag{7.17g}$$

$$t^s_i \leq T^{in\,max}_{i,j} + T^{max} \cdot (1 - a^s_{i,j}) \qquad \forall s \in S,\ (i,j) \in Ch(E) \tag{7.17h}$$

7.3 Application Example

In this section, a computational example for a fictional industrial cooling application is presented in order to demonstrate the approach. Instead of examining single components, the focus is on the interconnection and interaction of multiple components under varying loads, similar to the design of booster stations in the case of fluid systems. For this purpose, the system to be optimized is first described in more detail and subsequently the computational example is discussed.

7.3.1 Application to Industrial Cooling Systems

An interconnection of multiple chillers is called a chiller bank. A common configuration for such mutli-chiller systems with varying cooling demand is known as the primary-secondary system, sometimes also referred to as decoupled system. An exemplary schematic diagram is shown in Figure 7.3. The key benefits of this widely used closed-loop configuration are its simple control compared to other system configurations and its mostly fail-safe operation (Taylor, 2002). According to its name, a primary-secondary system is split into a primary side (generation side) and a secondary side (distribution side) using a bypass pipe, sometimes also called “decoupler”. The volume flow rate on the primary side is constant with respect to the active chillers. The chillers are connected in a parallel arrangement and typically each chiller is equipped with a dedicated constant-speed pump. In contrast, the volume flow rate on the secondary side is variable and can be adjusted using valves at the loads. Therefore, the secondary side is equipped with a variable-speed pump allowing to operate at varying volume flow rates. For the design of such arrangements, the primary volume flow rate is expected to be always greater than or equal to the secondary flow rate. The opposite scenario is excluded because it would occur if the system could not meet the demand. If the primary flow equals the secondary flow, the system is balanced. Otherwise, the bypass pipe is used to balance both sides. In this scenario, the excess chilled water produced mixes with the water leaving the loads before returning to the chillers.

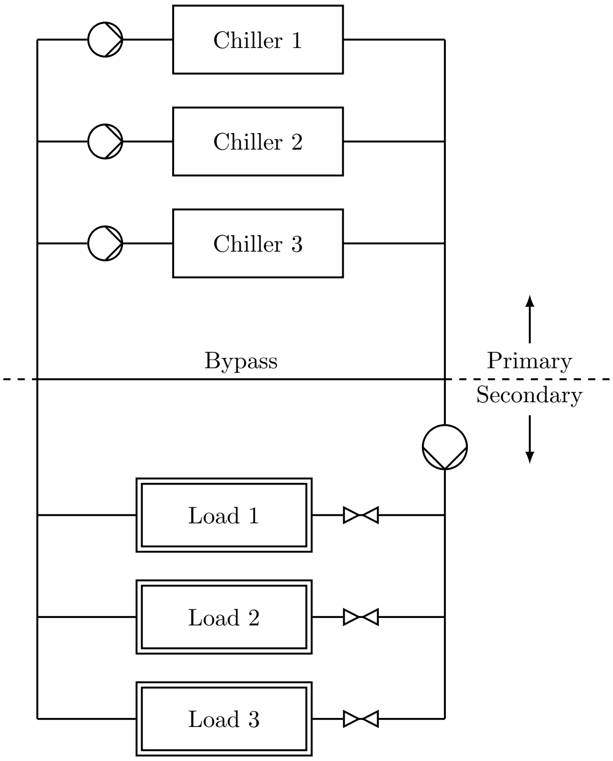


Figure 7.3: Illustration of a primary-secondary cooling system (based on Chang, 2004)

For demonstrating purposes, the following considerations are restricted to the thermal aspects of the system. Hence, the distribution aspect is only implicitly examined, e.g. pumps are not explicitly considered for the sake of simplicity. This is a common simplification and pumps as well as other auxiliary equipment can be either accounted for by an empirical function based on the expected system load or they can be ignored because of their inflexibility and comparably small influence on the overall costs (Risbeck et al., 2015). For the system considered here, this is particular true since the constant-speed pumps of the primary side are dedicated to individual chillers and therefore their on/off status and the operation of the variable-speed pump of the secondary side directly relates to the expected demand, which is assumed to be known a priori for the synthesis. Nevertheless, if necessary, the inclusion is straightforward if the corresponding characteristics of the system, e.g. pressure losses and pump characteristics, are known and the increased computational effort is expected to be reasonable. Alternatively, it is possible to integrate these decisions in detail as a subsequent step. This would, however, to some extend undermine the desired integrated approach. As a consequence, it should be reviewed in advance if the subsequent inclusion is suitable for the considered application.

The loads, which feed heat into the industrial cooling system, can be modeled as a lumped heat source. Hence, all thermal loads resulting from individual processes or machines are aggregated and represented as one large overall load. This is a reasonable representation since the chiller bank supplies a certain amount of chilled water at a certain temperature to the loads and the same amount of chilled water is returned to the chiller bank as a combined flow. Furthermore, as it is common in practice with the selected system design, it is not only assumed that the return temperature, due to the parallel arrangement, is the same for all chillers but also that all active chillers provide the same supply temperature (Kapoor, 2013). Therefore, the temperature of the chilled water entering the load is assumed to be known and the supply temperature becomes a constant parameter.

7.3.2 Computational Example

The considered optimization task involves a complete redesign of the chiller bank of an existing system. This means that no chillers are installed. However, the overall system design scheme, i.e. the primary-secondary arrangement, is not in question. For the computational example, a maximum thermal load of 5000 kW is assumed. Based on this, load scenarios with associated time shares are derived. A key characteristic for industrial sites, as considered here, is that the thermal load originates primarily from production processes and therefore permanent and year-round cooling is required. Accordingly, five different load scenarios that occur throughout the system's operational lifetime are considered, as summarized in Table 7.2.

The individual chillers of the chiller bank are modeled using the chiller model presented at the beginning of the chapter. According to this model, the volume

Table 7.2: Summary of expected thermal loads $\dot{Q}$, required volume flows $\dot{V}$ and the relative time shares F for the different load scenarios of the computational example

Scenario	$\dot{Q}$ $[kW]$	$\dot{V}$ $[m^3/h]$	F
1	5000	342	0.125
2	4250	306	0.250
3	3500	270	0.250
4	2750	234	0.250
5	2000	198	0.125

flow and supply temperature of a chiller are parameters and the return temperature changes with respect to the system loads. Hence, it is assumed that each chiller supplies chilled water at a fixed temperature of 5°C in all load scenarios. Furthermore, it is specified that the return temperature has to be in the range between 12° and 18°C to ensure efficient operation of the chillers, also taking into account operating limits. Building on these basic assumptions, four different exemplary chiller types are considered for the computational example. A summary of the parameters used for the individual chiller types is shown in Table 7.3. In this context, it is assumed that the power consumption of the chillers also comprises auxiliary equipment. Figure 7.4 illustrates the COP, i.e. the ratio between the cooling provided and the power required, as a function of the return temperature for the considered chiller types. As can be seen, efficiency increases with increasing return temperature since a reduced return temperature is equivalent to part-load operation. Furthermore, depending on the return temperature, different chiller types are most efficient. This representation based on the return temperature can be used as an alternative to using the difference between the supply and return temperature here since the supply temperature is fixed and assumed to be the same for each chiller and thus the efficiency of the chillers can be expressed by the return temperature alone.

Table 7.3: Nominal cooling capacities $\dot{Q}^N$, design volume flow rates $\dot{V}^N$ and regression parameters α, β of the available chiller types

Types	$\dot{Q}^N$ $[kW]$	$\dot{V}^N$ $[m^3/h]$	α $[kW/°C]$	β $[kW]$
C1	750	49.68	4.27	59.85
C2	850	56.27	7.72	41.31
C3	1500	99.32	10.41	106.63
C4	1750	115.88	17.51	56.91

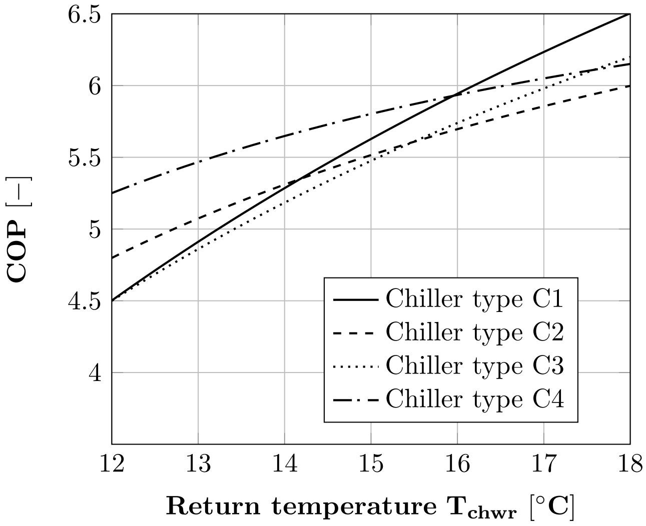


Figure 7.4: Overview of the COP as a function of return temperature T_{chwr} at constant supply temperature $T_{chws} = 5$°C for the available chiller types

The initial investment for each chiller type is determined based on the nominal cooling capacity using an empirical formula proposed by Bahl et al. (2016):

$$C^{buy}\big(\dot{Q}^N\big) = 0.8102 \cdot \dot{Q}^N \cdot \Big(179.63 + 4991.3436 \cdot \big(\dot{Q}^N\big)^{-0.6794}\Big) \tag{7.18}$$

In the context of this formula, $\dot{Q}^N$ [kW] is the nominal cooling capacity of a chiller and $C^{buy}\big(\dot{Q}^N\big)$ [€] is the investment as a function of the nominal cooling capacity. Furthermore, it is assumed that for simplicity the investment also includes pro rata costs for auxiliary equipment.

For the considered system, it is specified that a maximum of two chillers can be installed for each of the four chiller types. Hence, eight chillers are available (C1.1, C1.2, C2.1, ..., C4.2). The electricity costs are estimated at €0.16 per kWh. It is assumed that the system is designed for an operational lifetime of ten years with an average of 250 operating days per year and two shifts of eight hours per day.

7.3.3 Computational Results

In the following, the obtained results are presented. The runtime was in the range of seconds and is therefore not discussed further. The studied example is rather intended to demonstrate the approach as an academic example and acts as a proof of concept that provides a baseline for further investigations.

According to the optimal solution, four of the available eight chillers are to be purchased. These are each both available chillers of the chiller types two and four. This results in expected total costs of €4 822 717. As anticipated for an energy-intensive application such as cooling, the estimated energy costs of €3 907 064 far exceed the investment of €915 653. However, the investment still represents about 19% of the total costs.

Table 7.4: Overview of the active chillers, return temperatures T_{chwr} and volume flows through the decoupler $\dot{V}_{dec}$ for the different load scenarios of the computational example

Scenario	Active chillers	T_{chwr} [°C]	$\dot{V}_{dec}$ $[m^3/h]$
1	C2.1, C2.2, C4.1, C4.2	17.47	2.30
2	C2.1, C2.2, C4.1, C4.2	15.66	38.30
3	C2.2, C4.1, C4.2	15.46	18.04
4	C2.2, C4.1, C4.2	13.21	54.04
5	C4.1, C4.2	12.39	33.77
Total	C2.1, C2.2, C4.1, C4.2		

As shown in Table 7.4, the installed chillers are used in such a way that the flow through the decoupler is minimized and thus the temperature in the return is maximized because the efficiency of a chiller is assumed to increase with the return temperature. In this way, as the demand decreases, the chillers are gradually deactivated to better match the demand. In the first two load scenarios, all four installed chillers are used. In the third and fourth scenario, one of the previously active type two chillers is switched off due to decreasing demand. In the fifth scenario, only both larger chillers are used. Hence, the larger chillers of type four are used to cover the system's base load, while the smaller chillers are activated during peak load periods.

The obtained results underline that there is a need for reliable guidance since not only the efficiency of the individual chillers themselves together with other constraints, e.g. operating limits and investment, have to be considered simultaneously when deciding which chillers to install and activate but also the influence of these decisions on the efficiency of other chillers in the system.

8 Discrete-Time Representation for Thermofluid Systems

In the previous considerations, it was assumed that similar loads at different points in time can be aggregated to so-called load scenarios because the state of a system at a certain point in time did not depend on the temporal sequence, i.e. the state of the system was independent of the state at neighboring points in time. However, this approach is not applicable when chronology for adjacent points in time is required and the associated system states are therefore coupled. This is usually the case when considering storage components, components with extensive start-up and run-down phases, generally delayed system responses, or the like. The approaches and explanations presented in the following focus on the utilization of storage components, particularly thermal energy storage, although they may also be adaptable for other purposes.

As discussed in Section 2.2.4, the requirement of chronology increases the "model complexity" (see Bahl et al., 2018) compared to the previous considerations. Therefore, the use of an appropriate time representation is a crucial point. According to Floudas and Lin (2005), two main categories exist in literature—discrete-time and continuous-time representations. Discrete-time representations, which are the most widely used approach in literature, are based on time discretization. The observation period is thus divided into an a priori fixed number of time steps with a uniform length. All decisions to be made for the system, e.g. switching of components, and also changes in external conditions, e.g. changing loads, are associated with the start of a time step and can only be considered at these points in time. The advantages of this representation are mainly the straightforward implementation and simple formulation, which typically leads to well-structured optimization problems (Floudas and Lin, 2005).

Due to its simplicity and widespread use, a discrete-time representation is used in this chapter to extend the considerations to systems with storage components. Later, in Chapter 9, a continuous-time approach is provided as an alternative.

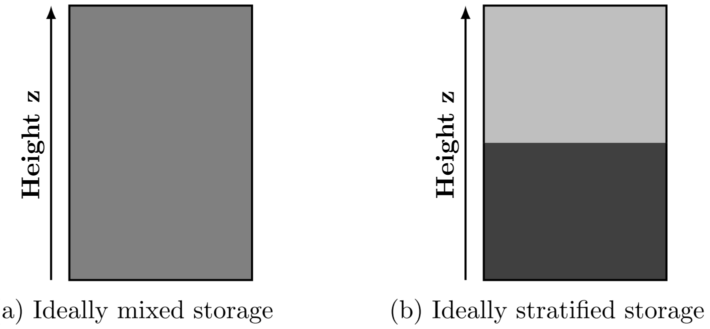
8.1 Thermal Energy Storage

For heating and cooling applications, the term “storage” usually correlates to storing thermal energy. In this context, thermal energy storage (TES) is used to decouple thermal demand and generation in order to compensate for fluctuations. The task of TES is therefore to store surplus thermal energy with as little loss as possible and to make it available as needed at a later time. This can improve the overall efficiency of a system as capacity requirements as well as operational costs of the generation side can be reduced (Walmsley et al., 2009). However, this is opposed by an increased complexity of the system and its control, investment for the TES tank and auxiliary equipment as well as potential storage losses.

Different technologies exist in the field of TES. In this work, the considerations are focused on storage tanks filled with liquids. A common example is the storage of cold water in large tanks for later use in industrial cooling systems. This type of TES falls into the class of so-called sensible heat storage, which is characterized by the fact that the storage medium only changes its sensible temperature, i.e. no phase transitions take place. Also, only TES without internal (built-in) heating or cooling components is considered.

In this regard, two ideal tank models with opposite characteristics can be distinguished in general. These are ideally mixed and ideally stratified tanks. Both are shown schematically in Figure 8.1. The ideally mixed tank has a homogeneous temperature, whereas the ideally stratified tank has two temperature zones of constant temperature (Baeten et al., 2014). Both variants are an idealized representation of the actual behavior. Depending on the design and operating conditions, however, the behavior can approach one or the other (Haller et al., 2009). In the following, stratified storage is assumed since stratification is often utilized in practice.

As indicated above, stratified tanks store fluid, e.g. water, simultaneously at two (or more) different temperatures. This is achieved by warmer water accumulating at the top of the tank due to the lower density and colder water accumulating at the bottom of the tank due to the higher density (Walmsley et al., 2009). The layers are separated from each other by an intermediate region, the so-called thermocline. This region is characterized by a sudden, steep change in temperature (Ma et al., 2009). Schematically, this is shown for an ideally stratified tank as a function of the position along the tank height in Figure 8.2. The actual position can move significantly up or down due to changes in thermal loads, such as variations in production rates or the operation of different production lines in industrial applications



(a) Ideally mixed storage (b) Ideally stratified storage

Figure 8.1: Illustration of different idealized storage models and associated temperature distribution along the height of the storage z, where darker colors represent colder fluid and lighter colors represent warmer fluid

(Walmsley et al., 2009). In the case of ideally stratified storage, the thermocline is infinitely thin. In practice, the rise is less steep and a thermocline of a certain thickness is formed, depending on how well the stratification can be achieved. In the following, however, it is assumed that the considered storage components are subject to a negligible interlayer mixing so that the idealized case is approximately achieved and thus represents a suitable simplification.

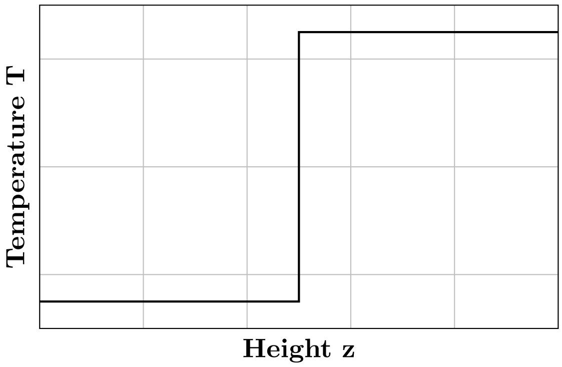


Figure 8.2: Schematic representation of the temperature profile and thermocline for an ideally stratified TES

For a two-layer stratified TES model, as considered here, warmer fluid, i.e. warmer than the thermocline, is assumed to be aggregated above the thermocline and colder fluid is assumed to be aggregated below the thermocline, resulting in a system that can be described by four states (Ma et al., 2009). The associated state variables are the temperatures of the warmer and colder layer and their associated height in the tank, or since it is expected that the geometry of the tank is known, the volumes of the two layers. The time-dependent description of these states technically involves differential equations. However, these are discretized in time, assuming constant conditions for the time steps, so that mathematical programming techniques can still be applied. Accordingly, a coupling of the respective states results for neighboring points in time. Depending on whether the tank is being charged or discharged, warmer fluid is stored in the top layer and colder fluid is withdrawn from the bottom layer or vice versa. For this purpose, it is assumed that the tank is completely filled with fluid at all times, with the sum of the volumes of the individual layers corresponding to the total volume of the tank. Accordingly, a change in volume of one of the layers, due to inflows or outflows during charging and discharging, causes a change in volume of the other layer to the same extent but with the opposite sign.

8.2 Extension of the Optimization Model for the Discrete-Time Representation

In the discrete-time representation considered in this chapter, there are no aggregated load scenarios as before since the temporal sequence of time steps is important because neighboring time steps are coupled by the storage levels of storage components. This is ensured by introducing the ordered set of time steps T. Thus, the set of load scenarios S is replaced by the time series T. However, this does not change the system behavior in general. Therefore, only the modified objective and the model extension for considering the new component type, i.e. a two-layer stratified thermal energy storage tank, are presented and described below. The remaining constraints can be obtained straightforwardly by substituting the load scenarios $s \in S$ used in the model for thermofluid systems from Chapter 7 with time steps $t \in T$. Furthermore, it is worth highlighting that it is not mandatory to use the presented TES model to apply the general approach. Instead, the TES model can be exchanged for other storage models if needed, e.g. to consider an ideally mixed tank or a stratified storage with layers of constant volume (see e.g. Schütz et al., 2015) since the general system behavior remains largely unchanged. All used sets, variables and parameters of the extension are shown in Table 8.1. If applicable, the dimension of an entry is specified using the base quantities mass (M), length (L), currency (C), time (T) and temperature (K).

Table 8.1: Variables, sets and parameters of the optimization model for thermofluid systems with TES using the discrete-time approach

Symbol	Range	Dimension	Description
$b_{i,j}$	$\{0,1\}$		Purchase indicator of edge $(i,j) \in E$
$a^t_{i,j}$	$\{0,1\}$		Activation indicator of edge $(i,j) \in E$ in time step $t \in T$
$p^t_{i,j}$	$\mathbb{R}^+_0$	ML^2T^{-3}	Power consumption of edge $(i,j) \in E$ in time step $t \in T$
$\dot{v}^t_{i,j}$	$\mathbb{R}^+_0$	L^3T^{-1}	Volume flow along edge $(i,j) \in E$ in time step $t \in T$
$\dot{u}^{in\ t}_{i,j}$	$\mathbb{R}^+_0$	ML^2T^{-3}	Thermal energy flow rate directly after the start of edge $(i,j) \in E$ in time step $t \in T$
$\dot{u}^{out\ t}_{i,j}$	$\mathbb{R}^+_0$	ML^2T^{-3}	Thermal energy flow rate directly before the end of edge $(i,j) \in E$ in time step $t \in T$
$\Delta h^t_{i,j}$	$\mathbb{R}$	$ML^{-1}T^{-2}$	Change in pressure caused by edge $(i,j) \in E$ in time step $t \in T$
v^t_a, v^t_b	$\mathbb{R}^+_0$	L^3	Stored volume of storage layer a and b for TES with edges $(a,b),(b,a) \in TES(E)$ in time step $t \in T$
$\Delta\dot{v}^t_a$, $\Delta\dot{v}^t_b$	$\mathbb{R}$	L^3T^{-1}	Net volume flow rate into storage layer a and b for TES with edges $(a,b),(b,a) \in TES(E)$ in time step $t \in T$
u^t_a, u^t_b	$\mathbb{R}^+_0$	ML^2T^{-2}	Thermal energy of storage layer a and b for TES with edges $(a,b),(b,a) \in TES(E)$ in time step $t \in T$
$\Delta\dot{u}^t_a$, $\Delta\dot{u}^t_b$	$\mathbb{R}$	ML^2T^{-3}	Net thermal energy flow rate into storage layer a and b for TES with edges $(a,b),(b,a) \in TES(E)$ in time step $t \in T$
t^t_a, t^t_b	$\mathbb{R}^+_0$	K	Temperature of storage layer a and b for TES with edges $(a,b),(b,a) \in TES(E)$ in time step $t \in T$
T			Ordered set of time steps
E			Set of edges
V			Set of vertices
$TES(E)$	$\subseteq E$		Subset of TES edges
$C^{buy}_{i,j}$	$\mathbb{R}^+_0$	C	Purchase costs of edge $(i,j) \in E$
$C^{kWh}_{i,j}$	$\mathbb{R}^+_0$	$M^{-1}L^{-2}CT^2$	Energy costs per kilowatt hour for edge $(i,j) \in E$
OLT	$\mathbb{N}_0$		Expected operational lifetime of the system, i.e. number of repetitions of time series T
$\Delta\tau^t$	$\mathbb{R}^+_0$	T	Length of time step $t \in T$
$\dot{V}^{max}$	$\mathbb{R}^+_0$	L^3T^{-1}	General upper bound on the volume flow
$V^{TES}_{a,b}$	$\mathbb{R}^+_0$	L^3	Total volume of TES with edges $(a,b),(b,a) \in TES(E)$
$\mathcal{T}(*)$	$\mathbb{R}^+_0$	K	Representation of the nonlinear temperature relationship

8.2.1 Objective Function

The Objective (8.1) remains largely unchanged. However, time steps are now considered instead of load scenarios. Accordingly, a parameter for the length $\Delta\tau^t$ of a time step $t \in T$ is used for the determination of the energy costs (right-hand term). In addition, the operational lifetime parameter OLT no longer indicates the total operational lifetime of a system, but how often the time series T, which occurs e.g. daily, is repeated during the operational lifetime of the system.

$$\min \sum_{(i,j)\in E} \left(C^{buy}_{i,j} \cdot b_{i,j} \right) + OLT \cdot \sum_{t\in T} \left(\Delta\tau^t \cdot \left(\sum_{(i,j)\in E} C^{kWh}_{i,j} \cdot p^t_{i,j} \right) \right) \tag{8.1}$$

8.2.2 Thermal Energy Storage Constraints

The two-layer stratified TES, described by Constraints (8.2), is modeled as a special type of component since it has the ability to store fluid. Instead of only one edge, it is modeled using two edges, representing the operation in charging and discharging mode. Moreover, there is no direct flow along the edges from one vertex to the other, i.e. no flow from one layer to the other, because the contents of the two layers are separated by the thermocline and are assumed not to mix directly. Instead, a flow into one of the vertices results in an equal amount of flow leaving the other vertex, causing the thermocline to move along the height of the tank, i.e. increasing the volume of the inlet layer and decreasing the volume of the outlet layer, as illustrated in Figure 8.3.

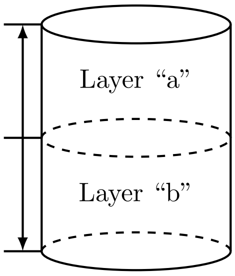


Figure 8.3: Illustration of thermocline movement for a two-layer stratified TES

If a TES tank is bought, both of its edges, representing the charge and discharge mode, are bought, see Constraint (8.2a). However, a TES tank cannot be charged and discharged at the same time, see Constraint (8.2b). Therefore, flow into or out of a respective layer can only occur if the corresponding mode, i.e. charging or discharging, is active, see Constraints (8.2c) to (8.2f). In other words, it is ensured that, for example, only colder fluid is taken from and warmer fluid is stored in the designated layer of a TES tank in discharge mode and vice versa in charge mode. As explained above, a TES tank is assumed to be always fully filled with fluid. Hence, the volume flow entering one layer is equal to the flow leaving the other layer. Accordingly, Constraint (8.2g) ensures that the sum of the volumes of both layers is always equal to the total volume of a TES tank. The net volume flow into a layer results from the difference of flows into and out of this layer, see Constraints (8.2h) and (8.2i). For this, at most one of the two flow terms for each layer is not equal to zero at the same time due to Constraints (8.2c) to (8.2f) described above. Depending on whether flow enters or leaves a layer, the net volume flow can be positive or negative. Both constraints are shown for the sake of a clearer presentation. However, the connection via $\Delta\dot{v}_a^t = -\Delta\dot{v}_b^t$ could also be used. The volume of each layer at time t depends on its volume at the previous time step $t-1$ and the net volume flow into the layer multiplied by the length of the time step, see Constraints (8.2j) and (8.2k). Since there is no predecessor for the first time step, initial charge levels need to be defined here. As before, one of the constraints would suffice, due to Constraint (8.2g), but both are stated explicitly for a more intuitive representation. The net thermal energy flow rate into a layer results from incoming and outgoing thermal energy flow rates, see Constraints (8.2l) and (8.2m). At most one of the two terms is non-zero and the net rate can be either positive or negative. In the presented case, no loss terms are considered. This is a reasonable simplification for small heat losses, e.g. for a very good isolation or short time intervals. Nevertheless, if necessary, additional terms can be added to account for losses to the environment or heat transfer between the two layers. However, these terms would be nonlinear in the case under consideration. This would require further linearizations, which in turn would lead to additional computational effort. Constraints (8.2n) and (8.2o) state that the thermal energy of a layer at time step t depends on the thermal energy of the previous time step $t-1$ and the net rate of thermal energy into that layer multiplied by the length of the time step. In addition, again initial conditions have to be defined since the first time step has no predecessor. As a special case, the temperature of a layer represented by the respective vertex is determined by its volume and thermal energy at time step t, taking into account the associated fluid properties, see Constraints (8.2p) and (8.2q). The nonlinear temperature relationship is modeled as before in the form of a piecewise linear approximation using the linearization techniques presented in Section 2.2.3. Potential pressure losses are neglected as a simplification for both the charge as well as the discharge mode, see Constraint (8.2r), but suitable loss terms can be added if required. Finally, TES components are not a powered component type themselves and therefore have no associated power consumption, see Constraint (8.2s).

$$b_{a,b} = b_{b,a} \qquad \forall\, (a,b) \in TES(E) \tag{8.2a}$$

$$a^t_{a,b} + a^t_{b,a} \leq 1 \qquad \forall\, t \in T,\ (a,b) \in TES(E) \tag{8.2b}$$

$$\sum_{(i,b)\in E: i\neq a} \dot{v}^t_{i,b} \leq (1 - a^t_{a,b}) \cdot \dot{V}^{max} \qquad \forall\, t \in T,\ (a,b) \in TES(E) \tag{8.2c}$$

$$\sum_{(a,j)\in E: j\neq b} \dot{v}^t_{a,j} \leq (1 - a^t_{a,b}) \cdot \dot{V}^{max} \qquad \forall\, t \in T,\ (a,b) \in TES(E) \tag{8.2d}$$

$$\sum_{(i,a)\in E: i\neq b} \dot{v}^t_{i,a} \leq (1 - a^t_{b,a}) \cdot \dot{V}^{max} \qquad \forall\, t \in T,\ (a,b) \in TES(E) \tag{8.2e}$$

$$\sum_{(b,j)\in E: j\neq a} \dot{v}^t_{b,j} \leq (1 - a^t_{b,a}) \cdot \dot{V}^{max} \qquad \forall\, t \in T,\ (a,b) \in TES(E) \tag{8.2f}$$

$$v^t_a + v^t_b = V^{TES}_{a,b} \qquad \forall\, t \in T,\ (a,b) \in TES(E) \tag{8.2g}$$

$$\Delta\dot{v}^t_a = \sum_{(i,a)\in E: i\neq b} \dot{v}^t_{i,a} - \sum_{(a,j)\in E: j\neq b} \dot{v}^t_{a,j} \qquad \forall\, t \in T,\ (a,b) \in TES(E) \tag{8.2h}$$

$$\Delta\dot{v}^t_b = \sum_{(i,b)\in E: i\neq a} \dot{v}^t_{i,b} - \sum_{(b,j)\in E: j\neq a} \dot{v}^t_{b,j} \qquad \forall\, t \in T,\ (a,b) \in TES(E) \tag{8.2i}$$

$$v^t_a = v^{t-1}_a + \Delta\dot{v}^t_a \cdot \Delta\tau^t \qquad \forall\, t \in T,\ a \in V : (a,b) \in TES(E) \tag{8.2j}$$

$$v^t_b = v^{t-1}_b + \Delta\dot{v}^t_b \cdot \Delta\tau^t \qquad \forall\, t \in T,\ b \in V : (a,b) \in TES(E) \tag{8.2k}$$

$$\Delta\dot{u}^t_a = \sum_{(i,a)\in E: i\neq b} \dot{u}^{out\,t}_{i,a} - \sum_{(a,j)\in E: j\neq b} \dot{u}^{in\,t}_{a,j} \qquad \forall\, t \in T,\ (a,b) \in TES(E) \tag{8.2l}$$

$$\Delta\dot{u}^t_b = \sum_{(i,b)\in E: i\neq a} \dot{u}^{out\,t}_{i,b} - \sum_{(b,j)\in E: j\neq a} \dot{u}^{in\,t}_{b,j} \qquad \forall\, t \in T,\ (a,b) \in TES(E) \tag{8.2m}$$

$$u^t_a = u^{t-1}_a + \Delta\dot{u}^t_a \cdot \Delta\tau^t \qquad \forall\, t \in T,\ a \in V : (a,b) \in TES(E) \tag{8.2n}$$

$$u^t_b = u^{t-1}_b + \Delta\dot{u}^t_b \cdot \Delta\tau^t \qquad \forall\, t \in T,\ b \in V : (a,b) \in TES(E) \tag{8.2o}$$

$$t^t_a = \mathcal{T}(v^t_a, u^t_a) \qquad \forall\, t \in T,\ a \in V : (a,b) \in TES(E) \tag{8.2p}$$

$$t^t_b = \mathcal{T}(v^t_b, u^t_b) \qquad \forall\, t \in T,\ b \in V : (a,b) \in TES(E) \tag{8.2q}$$

$$\Delta h^t_{a,b} = \Delta h^t_{b,a} = 0 \qquad \forall\, t \in T,\ (a,b) \in TES(E) \tag{8.2r}$$

$$p^t_{a,b} = p^t_{b,a} = 0 \qquad \forall\, t \in T,\ (a,b) \in TES(E) \tag{8.2s}$$

8.3 Application Example

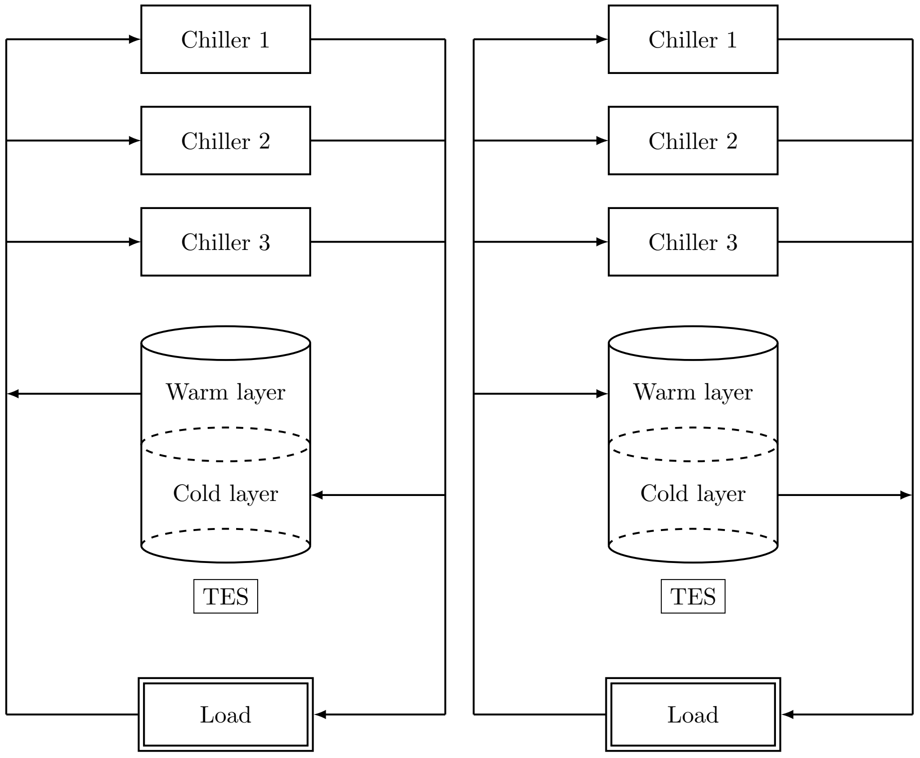
In this section, the computational example of a fictional industrial cooling application introduced in Section 7.3 is again used to demonstrate the approach. However, in contrast to the previous chapter, the integration of a TES tank, as described above, is considered. For this purpose, the general system structure is described first. Afterwards, the adapted computational example is examined and the differences to the system without storage are discussed.

8.3.1 Application to Industrial Cooling Systems

The basic system structure is similar to the previous case without storage and thus most explanations and assumptions given there also apply here. However, in the considered case, there is a TES tank that is integrated into the system so that the cooling no longer has to be provided just-in-time, but chilled water can be stored in the TES tank for later use. Accordingly, in case of excess production of chilled water, a certain amount is stored in the cold layer of the TES tank, forcing warmer water from the warm layer into the return of the system. Conversely, in case of a shortfall in production, chilled water is withdrawn from the cold layer of the TES tank and warmer water from the return is stored in the warm layer. Therefore, compared to the previous situation with the bypass pipe, both possible flow directions have to be considered. This is illustrated in Figure 8.4. As can be seen, depending on the direction of flow, the TES tank is either in charging or discharging mode.

8.3.2 Computational Example

In order to be able to compare the following computational example with the example examined in Section 7.3, the redesign of the chiller bank of an existing system is examined. Accordingly, no chillers are installed. However, a TES tank with known characteristics already exists for the system. Furthermore, in contrast to the previous example, chronological time steps have to be considered due to the consideration of storage. Therefore, eight discrete time steps with a fixed length of two hours each were set up such that the overall time series considered corresponds to the load scenarios from the previous example in terms of expected loads and frequency of occurrence. These are shown in Table 8.2. It is assumed that the time series under consideration represents one day with two eight-hour shifts, i.e. 16 hours of operation per day. In accordance with the example from Section 7.3,



(a) Charging of TES

(b) Discharging of TES

Figure 8.4: Illustration of a cooling system with stratified TES in charging and discharging mode (based on Deng et al., 2015)

the system is designed for an operational lifetime of ten years with an average of 250 operating days per year and estimated electricity costs of €0.16 per kWh.

With regard to the available chillers, a maximum of two chillers from each of the four chiller types introduced in Section 7.3 can be installed, i.e. eight chillers in total (C1.1, C1.2, C2.1, ..., C4.2). The performance characteristics and purchase costs of the chillers remain unchanged, see Table 7.3. Each chiller supplies chilled water at a fixed temperature of 5°C in all time steps. The return temperature is required to be in the range of 12° to 18°C to ensure efficient operation.

The already installed TES tank, which is intended to remain in operation, has a total volume of 235 m^3. For this, it is specified that the tank is completely discharged at the beginning of each day, i.e. at the beginning of the time series, which implies that the water in the tank has a uniform temperature of 18°C. Accordingly, it is assumed that the residual chilled water stored in the tank warms up

Table 8.2: Summary of expected thermal loads $\dot{Q}$ and required volume flows $\dot{V}$ for the different time steps of the computational example

Time step	$\dot{Q}$ $[kW]$	$\dot{V}$ $[m^3/h]$
1	2000	198
2	2750	234
3	3500	270
4	4250	306
5	5000	342
6	4250	306
7	3500	270
8	2750	234

overnight at the end of each day, i.e. at the end of the time series, so that the tank is completely discharged the next morning.

8.3.3 Computational Results

For the presented example, an optimal solution was found with the runtime being in the range of minutes. Therefore, the runtime is not discussed in detail for the considered example. However, further tests have shown that the runtime increases rapidly with a higher level of detail and larger instances.

According to the optimal solution found, three of the available eight chillers are installed. The installed chillers are one chiller of type two and both chillers of type four, as shown in Table 8.3. Thus, the topology in the considered case is close to the solution of the computational example without storage components from Section 7.3, see Table 7.4. In contrast, however, only one of the type two chillers is purchased. With the solution found, the installed cooling capacity can be significantly reduced compared to the case without storage. The maximum available cooling capacity of the optimal system is only 4350 kW, which is significantly lower than the maximum thermal load of 5000 kW. Thus, as expected, the decoupling of generation and demand can reduce the capacity requirements for the system.

Overall, the total costs are €4 592 041, of which 83.52% are energy costs. Accordingly, the expected energy costs are €3 835 207 and the investment is €756 834. Therefore, by using the TES tank, a total saving of €230 676 or 4.78% can be achieved compared to the system obtained in Section 7.3. It is noteworthy that 68.84% of this saving is due to the reduced initial investment. One reason for the comparatively low savings in energy costs can be attributed to the assumption that the storage is always fully discharged at the beginning of the time series, as

Table 8.3: Overview of the active chillers and TES modes for the different time steps of the computational example

Time step	Active chillers	TES mode
1	C4.1, C4.2	charge
2	C2.1, C4.1, C4.2	charge
3	C2.1, C4.1, C4.2	charge
4	C2.1, C4.1, C4.2	discharge
5	C2.1, C4.1, C4.2	discharge
6	C2.1, C4.1, C4.2	discharge
7	C2.1, C4.1, C4.2	charge
8	C4.1, C4.2	discharge
Total	C2.1, C4.1, C4.2	

this somewhat limits the degrees of freedom of the storage and tends to underestimate its benefits. However, for the approach considered, it is necessary to define the charging level at the beginning of the time series and the assumption made provides a plausible starting point.

The active chillers in each time step are shown in Table 8.3. It can be seen that all installed chillers are used during most time steps, except for the first and last time step when the smaller chiller is not active. This implies that the cooling provided by the chillers remains relatively constant due to the TES tank. Furthermore, Table 8.3 also gives an overview on the operating mode of the TES tank in each time step. For this, Figure 8.5 provides additional insight by plotting the charging level of the TES tank as a function of time. As can be seen, the TES tank is almost fully charged in the first three time steps due to the comparatively low demand. The stored chilled water is then used in the following three time steps with comparatively high demand for which the installed cooling capacity is not sufficient to meet the demand just-in-time. Finally, at the end of the time series, the TES tank is charged again in the penultimate time step and discharged in the last time step. The behavior in the last two time steps can be explained by the inflexibility of the chillers since it is not possible to match the chilled water generation exactly with the demand despite sufficient installed cooling capacity.

In addition, Figure 8.5 reveals that at the end of the time series, there is still an unused residual charge of about 25% for the TES tank, which dissipates overnight. The reason for this is that the discrete-time approach requires an a priori fixed time step length, which is problematic here because the system must operate in a specific operating mode for at least two hours. As a result, the benefits of the TES tank

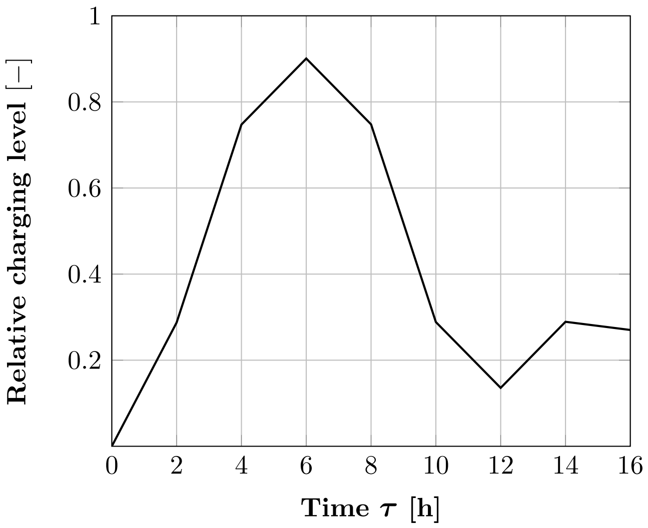


Figure 8.5: Charging level of the TES tank as a function of time for the computational example

cannot be fully exploited. Accordingly, this means that the theoretically achievable savings cannot be fully realized in the present example and, supplementing the above assumption, the obtained savings should rather be regarded as a conservative estimate.

9 Continuous-Time Representation for Thermofluid Systems

In general, discrete-time representations have two major disadvantages, as discussed in detail by Floudas and Lin (2005). Because of the a priori defined time steps with fixed, uniform length, a discrete-time representation is only an approximation of time, whose resolution depends on the number of time steps and their length. The length of the time steps is therefore often based on the smallest time period considered or on the greatest common divisor of all time periods since otherwise only sub-optimal solutions can be obtained. However, this leads to a large number of time steps, particularly for a short time step length or large observation periods, which in turn leads to large combinatorial problems. This is especially the case for real-world applications. Therefore, a considerable trade-off between accuracy and computational effort is typically required (Floudas and Lin, 2005).

Due to the drawbacks mentioned above, this chapter provides a continuous-time representation as an alternative approach. Continuous-time representations are generally based on the concept of variable time intervals represented by continuous timing variables. Therefore, it is possible that decisions to be made for the system as well as changes in external conditions can potentially take place at any time during the observation period. Compared to a discrete-time representation, a large fraction of redundant time steps can be eliminated, e.g. adjacent time steps for which both the control of the system and the external conditions do not change, but which are necessary because of the required uniform time resolution. This usually leads to smaller problems but is typically opposed by more complicated model structures (Floudas and Lin, 2005).

The challenges for the approach presented in the following are, on the one hand, estimating the number of variable time intervals needed and, on the other hand, dealing with the additional nonlinearities caused by the variable nature of the intervals compared to the discrete-time representation.

9.1 Estimating the Number of Intervals

The challenge in estimating and adjusting the number of time intervals is that while the length of the time intervals is variable, their number has to be determined a priori. If, on the one hand, the number of intervals is underestimated, inaccurate solutions or even infeasibility may result. On the other hand, if the number of intervals is overestimated, unnecessarily large instances arise. In the following, it is therefore shown how this challenge can be addressed for the considered scope.

To demonstrate the approach better, the example of a simple storage tank is used for illustration purposes. However, the application for other use cases, e.g. other storage components, is straightforward. In general, the change of the filling level for the considered storage tank during a certain time interval $\Delta\tau$ is given by:

$$\Delta V = \int_{\tau}^{\tau+\Delta\tau} \dot{V}(\tau)\, d\tau \tag{9.1}$$

Assuming constant derivatives, i.e. constant flow, within this interval, the above equation simplifies to:

$$\Delta V = \dot{V} \cdot \Delta\tau \tag{9.2}$$

In this context, there are two cases where the assumption of constant flows in the system for a considered time interval is not reasonable. The first case occurs when the external conditions, e.g. the loads, change. The second case occurs when the operation of the installed storage components changes, e.g. the start or end of a charging or discharging process, even if the external conditions are constant. Therefore, it is important to ensure that both the external conditions and the operation of the storage can be assumed to be constant during the time intervals considered.

As before, it is assumed that the timings and thus also the number of changes in external conditions are known in advance. In the following, the associated time intervals are referred to as time steps, following the terminology used in the discrete-time representation. What is not known in advance, however, are the timings and number of possible changes within these time steps that are caused by storage components. For this purpose, each time step with constant external conditions is divided into a certain number of so-called sub-intervals with variable length. Hence, the remaining challenge is to estimate the number of sub-intervals during a time step reasonably.

Key Assumption. *If there are constant external conditions, a storage component should strive to discharge as early as possible and to charge as late as possible during this time step to avoid energy losses. Even if potential energy losses are not explicitly considered, in many cases, it is reasonable to assume that the charging or discharging takes place in a continuous process instead of multiple, interrupted processes. As a suitable technical simplification, it is therefore assumed that for each storage component, there is the possibility for charging and discharging this component in each time step.*

Accordingly, in a system with one storage component, three cases can occur: The storage component can be charged, discharged or the system can cover the load without using the storage component. In the case of multiple storage components, two additional cases are considered for each additional storage component, representing the charging and discharging of that storage component. This provides a reasonable starting point for the estimation of the number of required sub-intervals per time step, which can usually be significantly reduced due to the assumptions made. With this, the number of sub-intervals considered ($n_{sub-intervals}$) is related to the number of storage components ($n_{storage}$) and an upper bound for the estimation of this number is given by:

$$n_{sub-intervals} = 2 \cdot n_{storage} + 1 \tag{9.3}$$

The general approach is illustrated in Figure 9.1 using a system with one simple storage tank that is assumed to be fully charged at the beginning of the observation period. In Figure 9.1, the charging level of the tank is plotted as a function of time, with the entire observation period divided into three consecutive time steps $\Delta\tau$ with constant external conditions during each time step. As discussed above, three sub-intervals for each time step are considered. For the first time step ($\Delta\tau_1$), all three sub-intervals are actually used: one for discharging the component ($i_{1,1}$), one while there is no change for the component ($i_{1,2}$), and one for charging the component ($i_{1,3}$). In contrast, for the second time step ($\Delta\tau_2$), it can be seen that the number of estimated sub-intervals for a time step is only an upper bound. While all three sub-intervals are used in the first time step, one sub-interval ($i_{2,1}$) would be sufficient for the second time step because the storage component is neither charged nor discharged during this time step. However, since this cannot be known in advance, three sub-intervals are nevertheless assumed here and assigned to the time step, with the remaining two sub-intervals not shown ($i_{2,2}$, $i_{2,3}$) consequently having a length of zero. A similar situation is shown for the third time step ($\Delta\tau_3$), where two of the three available sub-intervals ($i_{3,1}$, $i_{3,2}$) are used, while the remaining third sub-interval ($i_{3,3}$) has a length of zero.

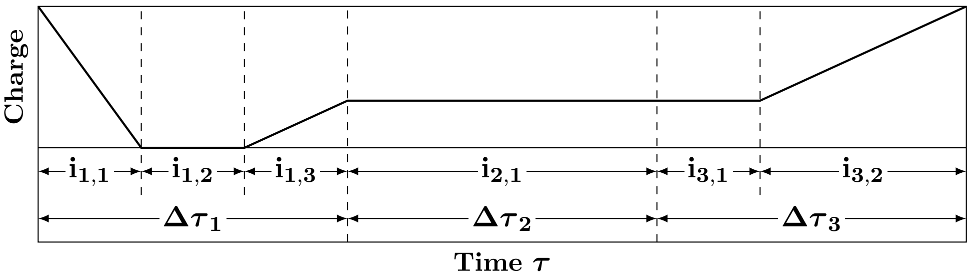


Figure 9.1: Illustration of the number of time steps and sub-intervals for exemplary charging and discharging processes of a storage component

With the presented approach, the number of time intervals increases only linearly with the number of storage components. Compared to the discrete representation with a similar resolution, this results in a smaller problem size because the resolution does not depend on the length of a predefined time step. Instead, time intervals can be potentially associated with any point in time where a change occurs in the system. Although it is not universally applicable, there are important applications in the field of fluid and thermofluid systems: thermal energy storage, storage tanks, and also delay effects resulting from long pipe lengths if these pipes are considered as a kind of storage component.

However, there are exceptions where the approach has to be modified considerably or cannot be used at all. One such exception is component cycling. This behavior, which is undesirable in many cases, can occur, for example, if a component on the generation side is significantly oversized and a storage component is therefore used to smooth its output. In this case, the component is switched on and off in short intervals multiple times and the storage is fully charged and discharged again in short succession. In the following, it is therefore assumed that the storage components do not aim at this kind of immediate smoothing of generation, but the storage is intended to cover deviating loads over time. Another exception are very long time steps with constant external conditions. In this case, additional sub-intervals are necessary, as otherwise the storage components might need to be significantly oversized in order to potentially meet the demand. In general, the number of estimated time steps should therefore be verified in case of doubt and adjusted if necessary. One possibility here is to adopt an iterative approach and successively increase the number of sub-intervals until no further improvement can be observed.

9.2 Alternative Linearization Approach

In general, the linearization techniques presented in Section 2.2.3 can also be used for the nonlinearities resulting from variable time intervals. However, a tradeoff between the linearization error and computation time has to be made here. Therefore, in the following, an exact linearization approach is presented, which can be used alternatively if appropriate conditions prevail.

In this context, let the following nonlinear equation be given, where the value of w is obtained from the product of two variables x and y:

$$w = x \cdot y \tag{9.4}$$

If x is a non-negative, real-valued variable, i.e. $x \in \mathbb{R}_0^+$, and y is a non-negative, integer variable, i.e. $y \in \mathbb{N}_0$, that can be bounded by a (reasonably large) number Y with $y \in \{0, \dots, Y\}$, an exact linear reformulation exists. The underlying idea is that y can be substituted by introducing $Y + 1$ binary variables $(z_0, \dots, z_Y)$. Each z_i thus represents one of the possible values of y. Accordingly, only one z_i may be non-zero and y can be determined by the sum of the products of all z_i with the respective index i:

$$\sum_{i=0}^{Y} z_i = 1 \tag{9.5a}$$

$$y = \sum_{i=0}^{Y} z_i \cdot i \tag{9.5b}$$

$$w = \sum_{i=0}^{Y} x \cdot z_i \cdot i \tag{9.5c}$$

$$z \in \{0, 1\} \tag{9.5d}$$

Nevertheless, there is still a nonlinear relationship that arises from the multiplication of x and z_i. Yet, since every summand is now a real-valued, non-negative variable multiplied by a binary variable, this can be reformulated using a well-known modeling technique. Hence, an auxiliary variable u_i with $u_i \in \mathbb{R}_0^+$ is introduced for each $x \cdot z_i$:

$$u_i = x \cdot z_i \qquad \forall\, i \in \{0, \ldots, Y\} \tag{9.6}$$

Using a big-M formulation, it is then ensured that $u_i = x$ if $z_i = 1$ and $u_i = 0$ otherwise:

$$\begin{aligned} u_i &\leq x && \forall\, i \in \{0, \ldots, Y\} && (9.7a)\\ u_i &\leq z_i \cdot M && \forall\, i \in \{0, \ldots, Y\} && (9.7b)\\ u_i &\geq x - (1 - z_i) \cdot M && \forall\, i \in \{0, \ldots, Y\} && (9.7c)\\ u_i &\geq 0 && \forall\, i \in \{0, \ldots, Y\} && (9.7d) \end{aligned}$$

As can be seen, this is only a reasonable approach if the number of possible values for y is not too large since otherwise a large number of additional auxiliary variables and additional constraints have to be introduced, which is computationally expensive.

In the considered scope, this approach can be applied to bilinear terms where one variable is the length of a sub-interval. In this case, the length of a sub-interval can be represented by the product of y and an a priori defined step size parameter. The downside is that the general continuity of the time representation is lost since the length of a sub-interval can only be a multiple of the predefined step size. However, in contrast to the time steps of the discrete-time representation, the length of the sub-intervals is still variable. If the achievable resolution is sufficient or at least acceptable, this approach is a viable alternative to the piecewise linear approximation due to the exact linearization and the potentially smaller problem size, provided that the number of possible values for y is not too large.

9.3 Extension of the Optimization Model for the Continuous-Time Representation

As in Chapter 8, there are no aggregated load scenarios since the temporal sequence of time steps is important because neighboring points in time are coupled by the storage levels of storage components. However, in addition to the ordered set for the time steps T, another ordered set I needs to be introduced to represent the sub-intervals of variable length within the "main" time steps. This does not change the system behavior in general with one exception, i.e. it has to be ensured that the sum of the lengths of the sub-intervals for a time step corresponds to the length of this time step. Therefore, only this additional constraint as well as the modified objective function and the model extension for considering a two-layer thermal energy storage tank, as described in Chapter 8, are presented in the following. The remaining constraints can be obtained straightforwardly by substituting the load scenarios $s \in S$ used in the model for thermofluid systems from Chapter 7 with time steps $t \in T$ and sub-intervals $l \in I$. All used sets, variables and parameters of the extension are shown in Table 9.1. If applicable, the dimension of an entry is specified using the base quantities mass (M), length (L), currency (C), time (T) and temperature (K).

Table 9.1: Variables, sets and parameters of the optimization model for thermofluid systems with TES using the continuous-time approach

Symbol	Range	Dimension	Description
$b_{i,j}$	$\{0,1\}$		Purchase indicator of edge $(i,j) \in E$
$a_{i,j}^{t,l}$	$\{0,1\}$		Activation indicator of edge $(i,j) \in E$ in time step $t \in T$ and interval $l \in I$
$\Delta\tau^{t,l}$	$\mathbb{R}_0^+$	T	Length of interval $l \in I$ for time step $t \in T$
$p_{i,j}^{t,l}$	$\mathbb{R}_0^+$	ML^2T^{-3}	Power consumption of edge $(i,j) \in E$ in time step $t \in T$ and interval $l \in I$
$\dot{v}_{i,j}^{t,l}$	$\mathbb{R}_0^+$	L^3T^{-1}	Volume flow along edge $(i,j) \in E$ in time step $t \in T$ and interval $l \in I$
$\dot{u}_{i,j}^{in\ t,l}$	$\mathbb{R}_0^+$	ML^2T^{-3}	Thermal energy flow rate directly after the start of edge $(i,j) \in E$ in time step $t \in T$ and interval $l \in I$
$\dot{u}_{i,j}^{out\ t,l}$	$\mathbb{R}_0^+$	ML^2T^{-3}	Thermal energy flow rate directly before the end of edge $(i,j) \in E$ in time step $t \in T$ and interval $l \in I$
$\Delta h_{i,j}^{t,l}$	$\mathbb{R}$	$ML^{-1}T^{-2}$	Change in pressure caused by edge $(i,j) \in E$ in time step $t \in T$ and interval $l \in I$
$v_a^{t,l}$, $v_b^{t,l}$	$\mathbb{R}_0^+$	L^3	Stored volume of storage layer a and b for TES with edges $(a,b),(b,a) \in TES(E)$ in time step $t \in T$ and interval $l \in I$
$\Delta\dot{v}_a^{t,l}$, $\Delta\dot{v}_b^{t,l}$	$\mathbb{R}$	L^3T^{-1}	Net volume flow rate into storage layer a and b for TES with edges $(a,b),(b,a) \in TES(E)$ in time step $t \in T$ and interval $l \in I$
$u_a^{t,l}$, $u_b^{t,l}$	$\mathbb{R}_0^+$	ML^2T^{-2}	Thermal energy of storage layer a and b for TES with edges $(a,b),(b,a) \in TES(E)$ in time step $t \in T$ and interval $l \in I$
$\Delta\dot{u}_a^{t,l}$, $\Delta\dot{u}_b^{t,l}$	$\mathbb{R}$	ML^2T^{-3}	Net thermal energy flow rate into storage layer a and b for TES with edges $(a,b),(b,a) \in TES(E)$ in time step $t \in T$ and interval $l \in I$

Continued on next page

Table 9.1: Continued from previous page

Symbol	Range	Dimension	Description
$t_a^{t,l}$ $t_b^{t,l}$	$\mathbb{R}_0^+$	K	Temperature of storage layer a and b for TES with edges $(a,b),(b,a) \in TES(E)$ in time step $t \in T$ and interval $l \in I$
T			Ordered set of time steps
I			Ordered set of intervals
E			Set of edges
V			Set of vertices
$TES(E)$	$\subseteq E$		Subset of TES edges
$C_{i,j}^{buy}$	$\mathbb{R}_0^+$	C	Purchase costs of edge $(i,j) \in E$
$C_{i,j}^{kWh}$	$\mathbb{R}_0^+$	$M^{-1}L^{-2}CT^2$	Energy costs per kilowatt hour for edge $(i,j) \in E$
OLT	$\mathbb{N}_0$		Expected operational lifetime of the system, i.e. number of repetitions of time series T
D^t	$\mathbb{R}_0^+$	T	Duration of time step $t \in T$
$\dot{V}^{max}$	$\mathbb{R}_0^+$	L^3T^{-1}	General upper bound on the volume flow
$V_{a,b}^{TES}$	$\mathbb{R}_0^+$	L^3	Total volume of TES with edges $(a,b),(b,a) \in TES(E)$
$\mathcal{T}(*)$	$\mathbb{R}_0^+$	K	Representation of the nonlinear temperature relationship
$\mathcal{L}(*)$	$\mathbb{R}_0^+$		Representation of the nonlinear time relationship

9.3.1 Objective Function

The Objective (9.8) remains largely unchanged compared to the previous chapter. As mentioned above, not only time steps but also sub-intervals during these time steps are considered and thus the energy costs are summed up over all time steps and the associated sub-intervals. Therefore, in addition to $t \in T$, the corresponding variables also have another index $l \in I$, which indicates the respective sub-interval. Furthermore, $\Delta\tau^{t,l}$, which specifies the length of a sub-interval of a time step, is no longer a parameter as in the discrete-time representation but a variable. Hence, there is a nonlinear term that needs to be linearized, represented here in a simplified way by an expression of the form $\mathcal{L}(*)$. In the chosen representation, care was taken to indicate that the linearization is not performed for each component individually, but that a linearization needs to be performed only once for each sub-interval of each time step, resulting in a significantly lower number of computationally expensive linearizations. In contrast to the nonlinearities considered so far, it is also possible to use the alternative linearization approach presented above, provided that favorable conditions prevail, i.e. a coarser temporal resolution is acceptable.

$$\min \quad \sum_{(i,j)\in E} \left(C^{buy}_{i,j} \cdot b_{i,j} \right) + OLT \cdot \sum_{t\in T} \sum_{l\in I} \mathcal{L}\Big(\Delta\tau^{t,l}, \Big(\sum_{(i,j)\in E} C^{kWh}_{i,j} \cdot p^{t,l}_{i,j} \Big) \Big) \tag{9.8}$$

9.3.2 General System Constraints

As mentioned above, it has to be ensured that for each time step the sum of the variable sub-interval lengths assigned to this time step matches its total length. This is ensured by Constraint (9.9). Apart from the modification of the indices, this is the only necessary change for the model that affects the system as a whole.

$$\sum_{l\in I} \Delta\tau^{t,l} = D^t \qquad \forall\, t \in T \tag{9.9}$$

9.3.3 Thermal Energy Storage Constraints

Most of the constraints introduced in Chapter 8 to describe a two-layer stratified TES remain the same, except for the consideration of the modified indices. These constraints are grayed out but displayed for the sake of completeness. However, the more strongly modified constraints are discussed in the following.

The volume of each layer at sub-interval l of time step t depends on its volume at the previous sub-interval $l-1$ and the net volume flow into that layer multiplied by the length of the respective sub-interval, see Constraints (9.10j) and (9.10k). However, in the continuous-time representation, $\Delta\tau^{t,l}$ is a variable. Therefore, the corresponding term is linearized, indicated by $\mathcal{L}(*)$. Furthermore, the transitions between time steps have to be considered. Accordingly, the predecessor of the first sub-interval, i.e. $l = 1$, of a time step t is the last sub-interval, i.e. $l = |I|$, of the previous time step $t-1$, where $|I|$ is the cardinality of the set of sub-intervals I. As in the discrete-time representation, it is also necessary to define initial charge levels as predecessors for the first sub-interval of the first time step. Constraints (9.10n) and (9.10o) are similar to Constraints (9.10j) and (9.10k) and state that the thermal energy of each layer at sub-interval l for time step t depends on the thermal energy of the previous sub-interval $l-1$ and the net thermal energy flow rate into that layer multiplied by the length of the respective sub-interval. Once more, due to the variable interval length, additional nonlinearites occur, represented by $\mathcal{L}(*)$. In addition, the same considerations regarding the transition between time steps and initial conditions apply as above. As with the objective function, the alternative exact linearization approach can be used here if favorable conditions prevail.

$$b_{a,b} = b_{b,a} \qquad \forall\, (a,b) \in TES(E) \tag{9.10a}$$

$$a_{a,b}^{t,l} + a_{b,a}^{t,l} \leq 1 \qquad \forall\, t \in T,\ l \in I,\ (a,b) \in TES(E) \tag{9.10b}$$

$$\sum_{(i,b) \in E: i \neq a} \dot{v}_{i,b}^{t,l} \leq (1 - a_{a,b}^{t,l}) \cdot \dot{V}^{max} \qquad \forall\, t \in T,\ l \in I,\ (a,b) \in TES(E) \tag{9.10c}$$

$$\sum_{(a,j) \in E: j \neq b} \dot{v}_{a,j}^{t,l} \leq (1 - a_{a,b}^{t,l}) \cdot \dot{V}^{max} \qquad \forall\, t \in T,\ l \in I,\ (a,b) \in TES(E) \tag{9.10d}$$

$$\sum_{(i,a) \in E: i \neq b} \dot{v}_{i,a}^{t,l} \leq (1 - a_{b,a}^{t,l}) \cdot \dot{V}^{max} \qquad \forall\, t \in T,\ l \in I,\ (a,b) \in TES(E) \tag{9.10e}$$

$$\sum_{(b,j) \in E: j \neq a} \dot{v}_{b,j}^{t,l} \leq (1 - a_{b,a}^{t,l}) \cdot \dot{V}^{max} \qquad \forall\, t \in T,\ l \in I,\ (a,b) \in TES(E) \tag{9.10f}$$

$$v_a^{t,l} + v_b^{t,l} = V_{a,b}^{TES} \qquad \forall\, t \in T,\ l \in I,\ (a,b) \in TES(E) \tag{9.10g}$$

$$\Delta\dot{v}_a^{t,l} = \sum_{(i,a) \in E: i \neq b} \dot{v}_{i,a}^{t,l} - \sum_{(a,j) \in E: j \neq b} \dot{v}_{a,j}^{t,l} \qquad \forall\, t \in T,\ l \in I,\ (a,b) \in TES(E) \tag{9.10h}$$

$$\Delta\dot{v}_b^{t,l} = \sum_{(i,b) \in E: i \neq a} \dot{v}_{i,b}^{t,l} - \sum_{(b,j) \in E: j \neq a} \dot{v}_{b,j}^{t,l} \qquad \forall\, t \in T,\ l \in I,\ (a,b) \in TES(E) \tag{9.10i}$$

$$v_a^{t,l} = v_a^{t,l-1} + \mathcal{L}(\Delta\dot{v}_a^{t,l}, \Delta\tau^{t,l}) \qquad \forall\, t \in T,\ l \in I,\ a \in V : (a,b) \in TES(E) \tag{9.10j}$$

$$v_b^{t,l} = v_b^{t,l-1} + \mathcal{L}(\Delta\dot{v}_b^{t,l}, \Delta\tau^{t,l}) \qquad \forall\, t \in T,\ l \in I,\ b \in V : (a,b) \in TES(E) \tag{9.10k}$$

$$\Delta\dot{u}_a^{t,l} = \sum_{(i,a) \in E: i \neq b} \dot{u}_{i,a}^{out\ t,l} - \sum_{(a,j) \in E: j \neq b} \dot{u}_{a,j}^{in\ t,l} \qquad \forall\, t \in T,\ l \in I,\ (a,b) \in TES(E) \tag{9.10l}$$

$$\Delta\dot{u}_b^{t,l} = \sum_{(i,b) \in E: i \neq a} \dot{u}_{i,b}^{out\ t,l} - \sum_{(b,j) \in E: j \neq a} \dot{u}_{b,j}^{in\, t,l} \qquad \forall\, t \in T,\ l \in I,\ (a,b) \in TES(E) \tag{9.10m}$$

$$u_a^{t,l} = u_a^{t,l-1} + \mathcal{L}(\Delta\dot{u}_a^{t,l}, \Delta\tau^{t,l}) \qquad \forall\, t \in T,\ l \in I,\ a \in V : (a,b) \in TES(E) \tag{9.10n}$$

$$u_b^{t,l} = u_b^{t,l-1} + \mathcal{L}(\Delta\dot{u}_b^{t,l}, \Delta\tau^{t,l}) \qquad \forall\, t \in T,\ l \in I,\ b \in V : (a,b) \in TES(E) \tag{9.10o}$$

$$t_a^{t,l} = \mathcal{T}(v_a^{t,l}, u_a^{t,l}) \qquad \forall\, t \in T,\ l \in I,\ a \in V : (a,b) \in TES(E) \tag{9.10p}$$

$$t_b^{t,l} = \mathcal{T}(v_b^{t,l}, u_b^{t,l}) \qquad \forall\, t \in T,\ l \in I,\ b \in V : (a,b) \in TES(E) \tag{9.10q}$$

$$\Delta h_{a,b}^{t,l} = \Delta h_{b,a}^{t,l} = 0 \qquad \forall\, t \in T,\ l \in I,\ (a,b) \in TES(E) \tag{9.10r}$$

$$p_{a,b}^{t,l} = p_{b,a}^{t,l} = 0 \qquad \forall\, t \in T,\ l \in I,\ (a,b) \in TES(E) \tag{9.10s}$$

9.4 Application Example

In the following, the computational example of a fictional industrial cooling application is considered again. This time, however, the continuous-time approach described above is applied. To be able to show the approach, some simplifications are made compared to the previous example from Section 8.3. Furthermore, in order to put the results into perspective, the discrete-time approach is applied to the same example, i.e. also taking into account the same assumptions, and the differences are compared and discussed.

9.4.1 Computational Example

The general system structure and the considered optimization task do not change compared to the computational example for the discrete-time representation from Section 8.3, i.e. the redesign of the chiller bank of an existing system with a TES tank is examined. However, due to the higher computational effort associated with the introduction of variable sub-intervals, only four time steps, each with a length of two hours, are considered. These are shown in Table 9.2. Each time step is divided into three sub-intervals. In this context, the alternative linearization approach described above is used in order to limit the linearization error. Using this approach, a resolution of ten minutes is considered, which is assumed to be sufficient for the example under consideration. The time series repeats twice per day, which corresponds to the observation of two eight-hour shifts. The example is based on an operational lifetime of ten years and an average of 250 operating days per year as well as electricity costs of €0.16 per kWh.

Table 9.2: Summary of expected thermal loads $\dot{Q}$ and required volume flows $\dot{V}$ for the different time steps of the computational example

Time step	$\dot{Q}$ [kW]	$\dot{V}$ $[m^3/h]$
1	2375	216
2	3875	288
3	4625	324
4	3125	252

In this example, the installed TES tank has a total volume of 150 m^3. As a simplification, it is assumed that the water in the warm layer of the TES tank always has a constant temperature of 18°C. The residual charge is assumed to dissipate at the end of a time series, i.e. at the end of each shift. Hence, the water in the tank has a uniform temperature of 18°C at the beginning of each shift.

With regard to the available chillers, the possible options remain unchanged compared to the two previous examples. Hence, the same eight chillers (C1.1, C1.2, C2.1, ..., C4.2) with the same characteristics can be installed, see Table 7.3. Each chiller supplies chilled water at a fixed temperature of 5°C and the return temperature has to be in the range of 12° to 18°C to ensure efficient operation.

9.4.2 Computational Results

Within a computing time of six hours, no provably optimal solution could be found for the example at hand. The calculations were performed on a Macbook Pro Early 2015 with a 3.1 GHz Intel Core i7 and 16 GB 1867 MHz DDR3 memory using CPLEX Optimization Studio 12.10 as MILP solver. The remaining optimality gap was 9.98%. Further test runs revealed that even with an increased computation time of twelve hours, this gap could not be reduced. The following discussion therefore focuses on the best solution found, which is then compared to the optimal solution for the discrete-time approach with the same number of time steps.

In the continuous-time case, a solution with total costs of €4 507 926 was found. For this, the investment is €740 979 and the expected energy costs for ten years are €3 766 947. Accordingly, 16.44% of the total costs are attributable to the initial investment. Compared to the discrete-time approach, savings of €115 896 are expected. In this context, it is noteworthy that the topologies of both systems differ. While in the discrete-time case a chiller of the second type and both available chillers of the fourth type are purchased, a smaller type one chiller replaces the slightly larger type two chiller in the continuous-time case.

Table 9.3 shows the active chillers and the operating mode of the TES tank in the individual sub-intervals for the continuous-time approach. It can be seen that two of the three available sub-intervals are used in the first and last time step and only one sub-interval in each of the middle two time steps. Although the obtained solution is not the provably optimal solution, the assumption of using at most three sub-intervals seems to be suitable for the considered example. In the first time step, the TES tank is charged, initially using only the two larger chillers and later also activating the smaller peak-load chiller. Here, the first sub-interval of the first time step basically corresponds to the just-in-time generation of the required cooling since due to the inflexibility of the chillers an exact balancing of generation and demand is only possible in rare cases. In the next two time steps, the TES tank is discharged with all chillers being active. The utilization of the TES tank is necessary here because the installed chillers cannot provide the required cooling during these time steps. In the last time step, the TES tank is first charged again, with all chillers being active, and then finally discharged in the second sub-interval.

Table 9.3: Overview of the active chillers and TES modes for the different sub-intervals during the time steps of the computational example using the continuous-time approach

Time step	Interval	Duration [h]	Active chillers	TES mode
1	1	0.50	C4.1, C4.2	charge
	2	1.50	C1.1, C4.1, C4.2	charge
	3	-	-	-
2	1	2.00	C1.1, C4.1, C4.2	discharge
	2	-	-	-
	3	-	-	-
3	1	2.00	C1.1, C4.1, C4.2	discharge
	2	-	-	-
	3	-	-	-
4	1	0.67	C1.1, C4.1, C4.2	charge
	2	1.33	C4.1, C4.2	discharge
	3	-	-	-
Total			C1.1, C4.1, C4.2	

For the application of the discrete-time approach, in addition to the different topology mentioned above, there is also a different operating behavior. This is shown in Table 9.4. All chillers are active in all time steps except the last time step. Accordingly, the TES tank is charged in the first two time steps, with the charging level changing only marginally in the second time step, and discharged in the following two time steps.

Table 9.4: Overview of the active chillers and TES modes for the different time steps of the computational example using the discrete-time approach

Time step	Active chillers	TES mode
1	C2.1, C4.1, C4.2	charge
2	C2.1, C4.1, C4.2	charge
3	C2.1, C4.1, C4.2	discharge
4	C4.1, C4.2	discharge
Total	C2.1, C4.1, C4.2	

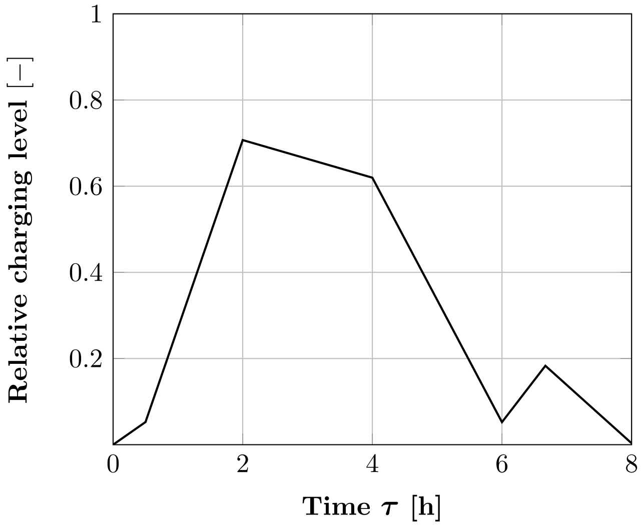


Figure 9.2: Charging level of the TES tank as a function of time for the computational example using the continuous-time approach

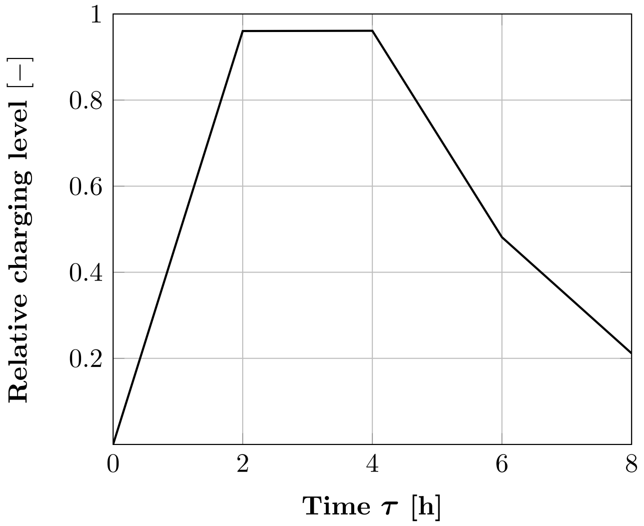


Figure 9.3: Charging level of the TES tank as a function of time for the computational example using the discrete-time approach

By comparing the different charging level profiles in Figures 9.2 and 9.3, two further observations can be made. First, by using the continuous-time approach, the TES tank is almost completely discharged at the end of the time series. Thus, most of the cooling provided is actually used, while the discrete approach results in a residual charge of about 20%, which is lost at the end of the time series. Second, for the continuous-time approach, a much smaller tank would suffice since the TES tank is only used to about 70% of its capacity. For the discrete-time approach, in contrast, the available storage capacity is almost completely utilized.

Overall, the example shows the considerable differences in the results of the two approaches and the benefits of the continuous-time approach. The variable time intervals lead, as expected, to lower energy costs, e.g. due to the more flexible control of the chillers and a better utilization of the stored chilled water. In addition, a different, less expensive topology for the overall system is found. Furthermore, for the example at hand, it would also be possible to replace the storage tank with a much smaller component. Taking these aspects into account, it becomes evident that despite the additional modeling and computational effort, the consideration of the continuous-time approach can be advantageous.

10 Conclusion and Outlook

The essence of this thesis was to make a contribution to the vision of providing decision support for engineers in order to supplement the human intuition during the design of technical systems. The focus was set on bridging engineering and algorithmic optimization and introducing the relevant methodology of algorithmic optimization for engineers in a purpose-oriented manner. Within this context, the optimization of fluid and thermofluid systems was studied. As a consequence of the extensive scope of the subject, the overall challenge was divided into individual sub-challenges. Selected sub-challenges were addressed in this thesis to provide substantial progress towards the overall vision and to establish a foundation for current and future work on technical applications in the field of fluid and thermofluid systems. In the following, the individual contributions of this thesis are concluded and future research directions are discussed.

10.1 Conclusion

Based on the introduction of a suitable modular basic optimization model for fluid systems, primal and dual heuristics were introduced and combined in a Branch-and-Bound framework. Using these heuristics that rely on domain-specific knowledge, it was possible to solve relatively large instances in reasonable time using the application example of booster stations. For this purpose, the instances were designed in such a way that they correspond to practice-relevant applications with varying loads represented by different load scenarios. With this approach, state-of-the-art MILP solvers, which could not solve the provided instances, were outperformed. The runtimes of the individual heuristics and the Branch-and-Bound framework were fast enough to be of practical relevance for the considered application.

Furthermore, a QMIP formulation was used in order to design cost-efficient resilient systems, exemplified by booster stations. Moreover, it could also be ensured that the resulting booster stations meet the additional requirements specified in DIN 1988-500. The presented approach can be directly integrated into the proposed workflow. In this context, it proved to have the potential to support system designers in two different ways. Firstly, increasing resilience is made easy. The system designer can focus on the main functionality, while the approach takes care of resilience. Existing non-resilient systems can be transformed into more resilient ones without questioning the initial system. Secondly, the approach helps to overcome smaller design flaws. On top of increasing resilience, it can help to save energy. This is also beneficial with regard to off-the-shelf systems as they can be made more resilient as well as adapted to the actual load conditions simultaneously. Thus, the presented approach combines resource-efficiency and reliability.

Finally, three model extensions for the synthesis of thermofluid systems were presented. In the first extension, it was shown how it is possible to integrate thermal considerations in order to be able to examine thermofluid systems but without compromising compatibility with the basic fluid system optimization model. The second and third extensions focused on the consideration of the time-dependent behavior of technical fluid-based systems that incorporate storage. Based on the introduction of thermal energy storage, a discrete-time representation as well as a continuous-time representation with variable time intervals were introduced. All three extensions were presented using the application example of an industrial cooling system. Furthermore, the two time representations were compared with each other. In this context, the continuous-time approach showed great potential for further consideration since it is possible to overcome a major part of the shortcomings of the traditional discrete-time approach.

10.2 Outlook

This thesis provides a foundation for the algorithmic synthesis of fluid and thermofluid systems. However, the overall vision to model the synthesis of fluid-based systems in a general and consistent way and at the same time to be able to perform algorithmic optimization in practice has not yet fully been realized as this is still far in the future due to its huge scope. Therefore, future research for the remaining sub-challenges is necessary to fulfill this visionary task.

The first step in this context is to further advance algorithmic investigations and to develop suitable solution procedures to solve large, practical instances efficiently. The results for the approach developed in this thesis to use heuristics with domain-specific knowledge in the field of fluid systems show that an adaptation of this approach for thermofluid systems seems promising. For thermofluid systems without storage components, this adaptation is expected to be quite straightforward. However, in the important case of the time-dependent formulation, it has to be examined to what extent the additional couplings of time steps, which result from the storage components, can be integrated in a reasonable way. In addition, further application-related algorithmic methods for the efficient solution of larger instances can be investigated, in particular with regard to the exploitation of the technical properties of thermofluid systems. Besides this, another important aspect is the development of efficient approaches for handling nonlinearities, especially with regard to the necessary number of linearizations and the utilization of alternative linearization techniques and approaches.

In a subsequent step, it is important to validate the developed models and methods by detailed simulations. With the help of this intermediate step, which precedes the practical application, the necessary level of detail of the obtained solutions can be evaluated. Thus, the models and methods can be further refined to ensure a smooth transfer into practice and reliable guidance for real-world systems.

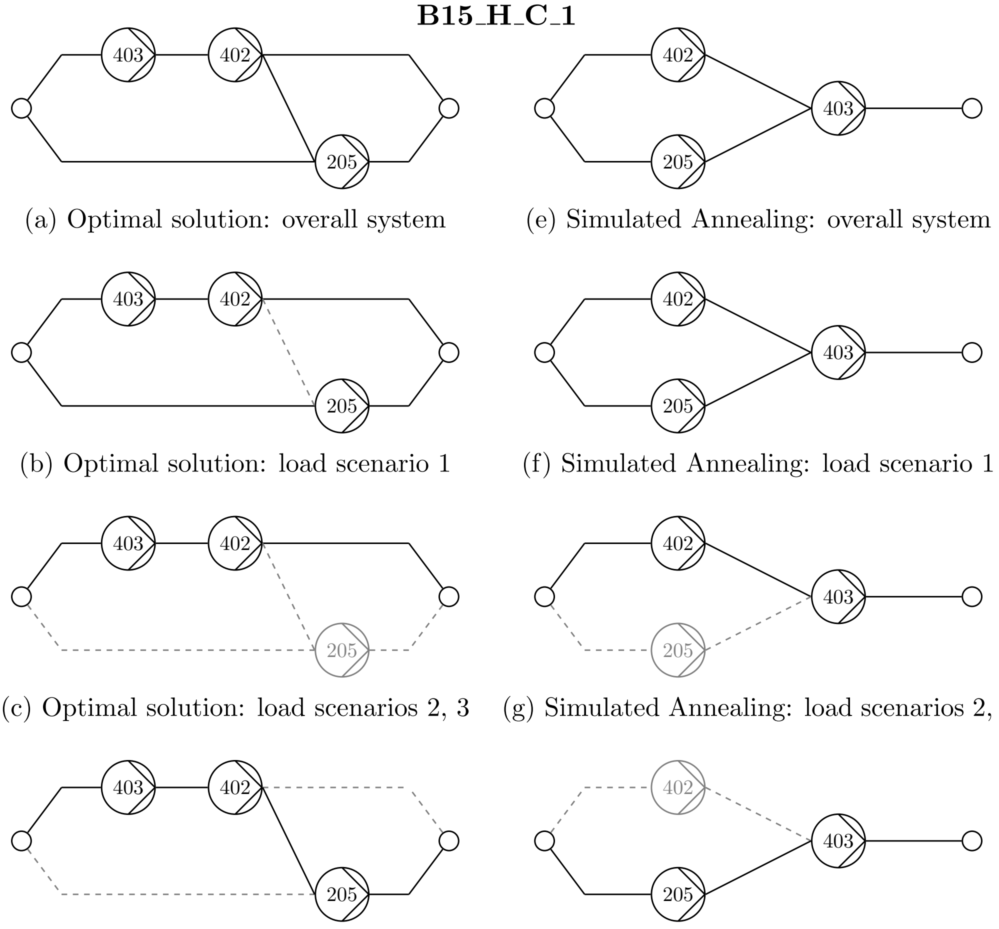
In addition, accessibility for practitioners is a consideration. Bundling the developed models and methods in a unified software framework could contribute to this. With this in mind, care has already been taken in this thesis to provide models and methods that are as comprehensive and broadly applicable as possible and that these are suitable for integration into a software tool for system designers. In this context, providing a suitable GUI could furthermore ensure that the application of the methodology is not tied to profound background knowledge, which in turn could increase acceptance.

Pursued further, there is a reasonable chance and perspective that algorithmic optimization as applied in this work could one day have the same importance for the design of technical systems as similar methods already have in the context of the optimization of production and logistics-related processes.

Appendix A

A.1 Overview of the Topological Results for Section 5.5

B15_H_C_1



(a) Optimal solution: overall system

(e) Simulated Annealing: overall system

(b) Optimal solution: load scenario 1

(f) Simulated Annealing: load scenario 1

(c) Optimal solution: load scenarios 2, 3

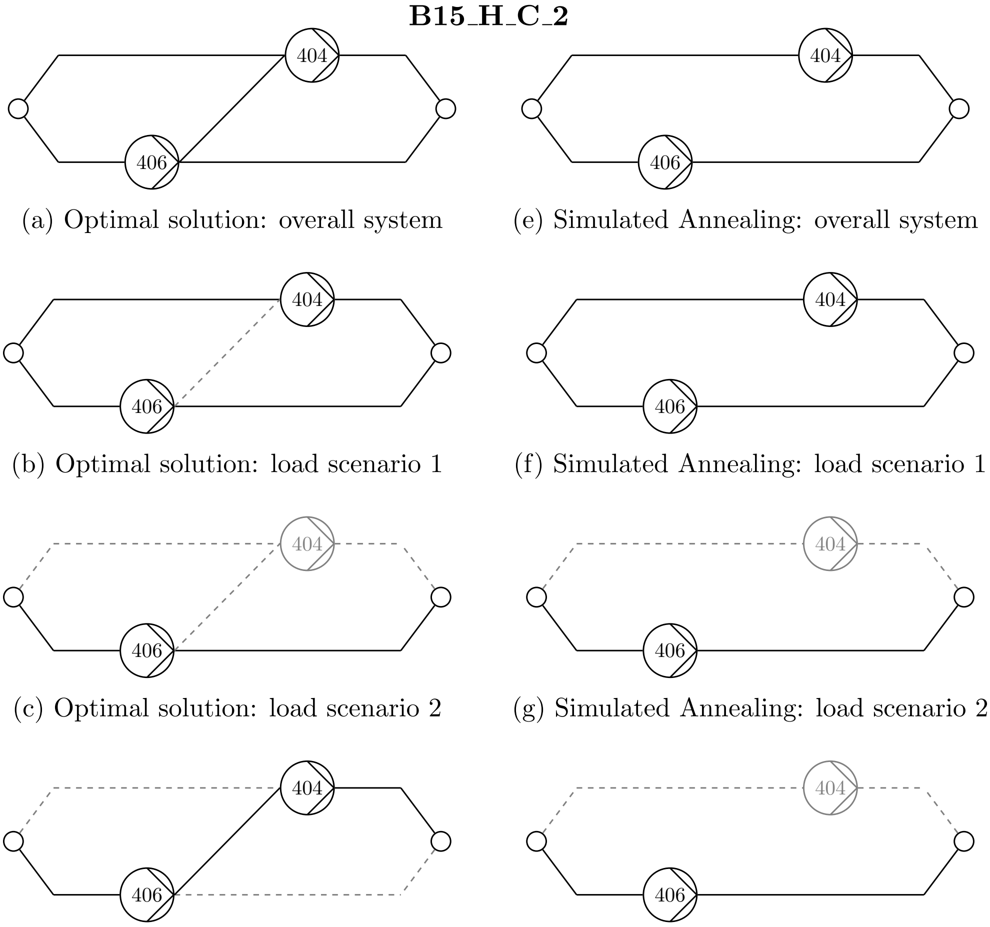
(g) Simulated Annealing: load scenarios 2, 3

(d) Optimal solution: load scenarios 4, 5

(h) Simulated Annealing: load scenarios 4, 5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B15_H_C_1

B15_H_C_2



(a) Optimal solution: overall system

(e) Simulated Annealing: overall system

(b) Optimal solution: load scenario 1

(f) Simulated Annealing: load scenario 1

(c) Optimal solution: load scenario 2

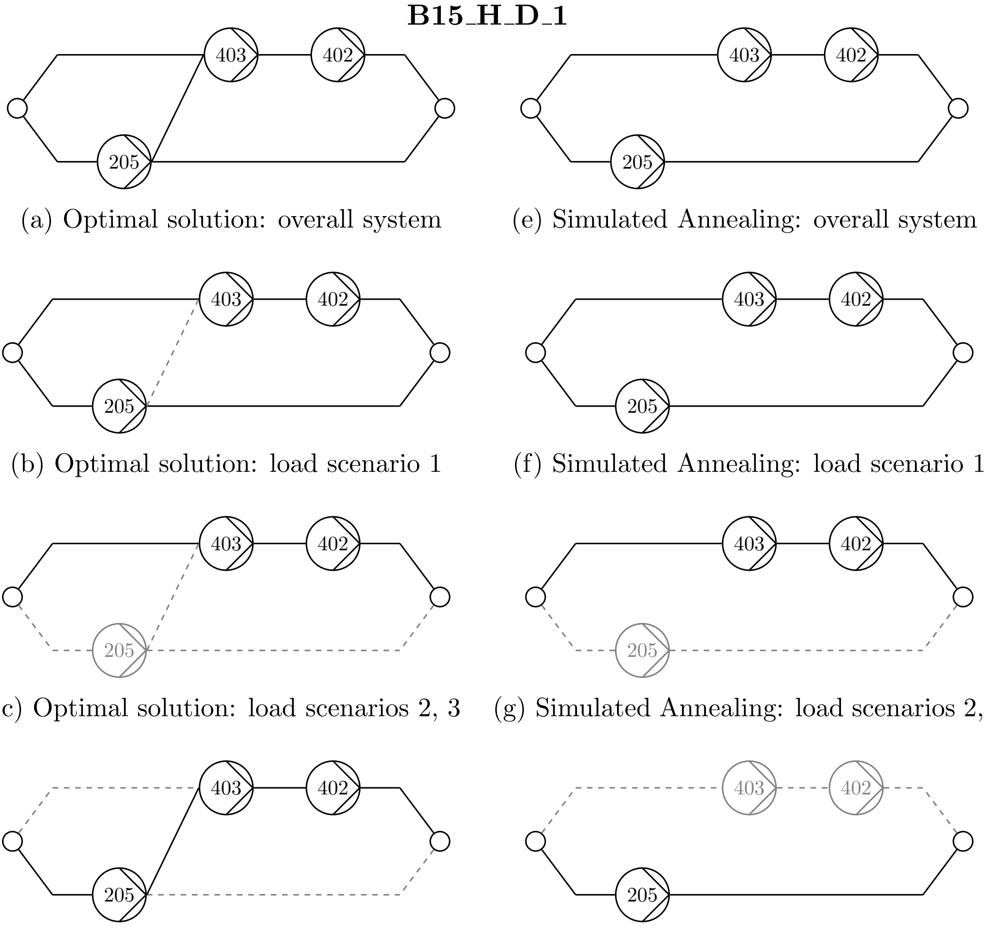
(g) Simulated Annealing: load scenario 2

(d) Optimal solution: load scenarios 3–5

(h) Simulated Annealing: load scenarios 3–5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B15_H_C_2

B15_H_D_1



(a) Optimal solution: overall system

(e) Simulated Annealing: overall system

(b) Optimal solution: load scenario 1

(f) Simulated Annealing: load scenario 1

(c) Optimal solution: load scenarios 2, 3

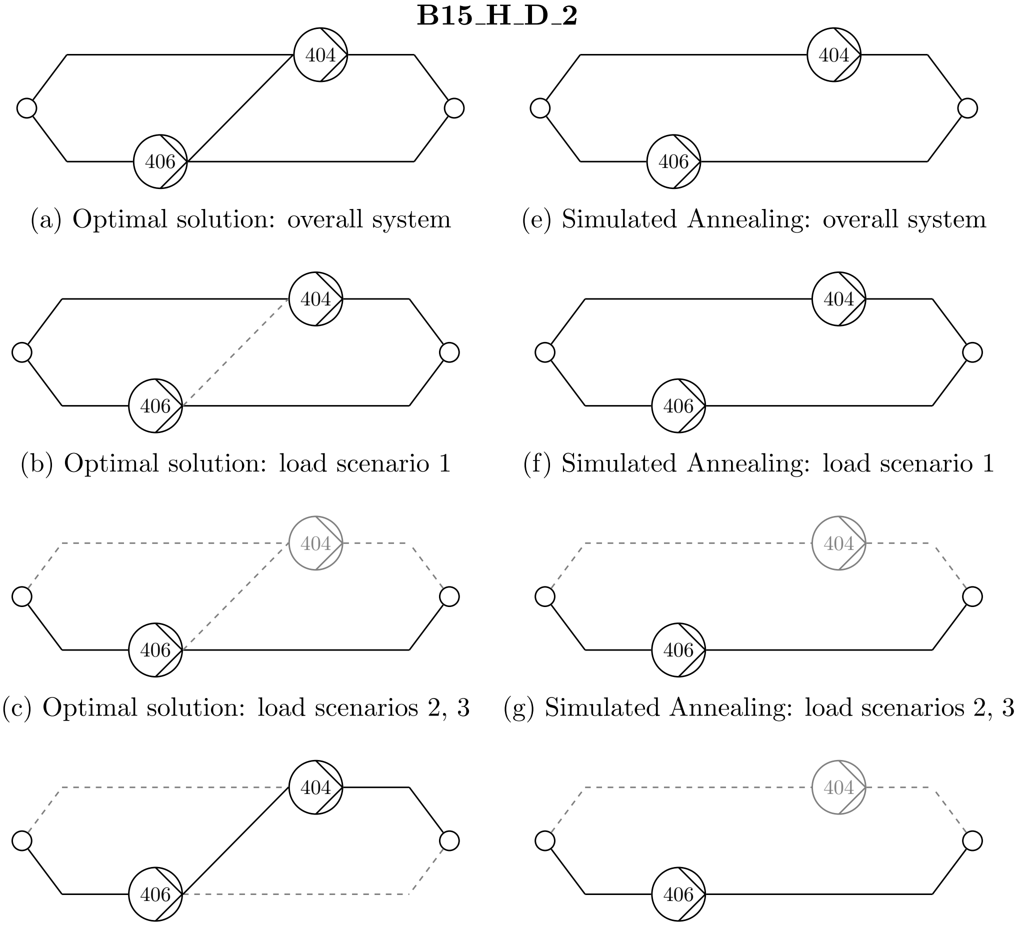
(g) Simulated Annealing: load scenarios 2, 3

(d) Optimal solution: load scenarios 4, 5

(h) Simulated Annealing: load scenarios 4, 5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B15_H_D_1

B15_H_D_2



(a) Optimal solution: overall system

(e) Simulated Annealing: overall system

(b) Optimal solution: load scenario 1

(f) Simulated Annealing: load scenario 1

(c) Optimal solution: load scenarios 2, 3

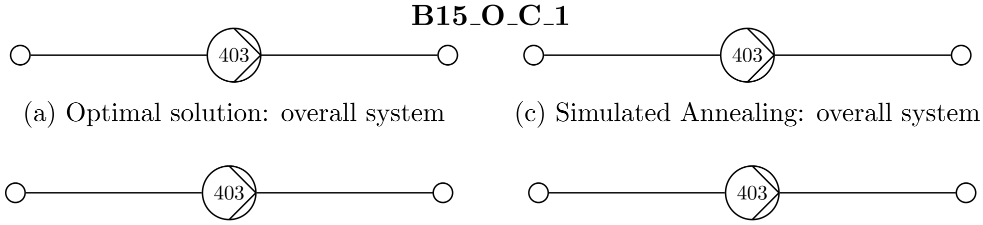
(g) Simulated Annealing: load scenarios 2, 3

(d) Optimal solution: load scenarios 4, 5

(h) Simulated Annealing: load scenarios 4,5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B15_H_D_2

B15_O_C_1



(a) Optimal solution: overall system

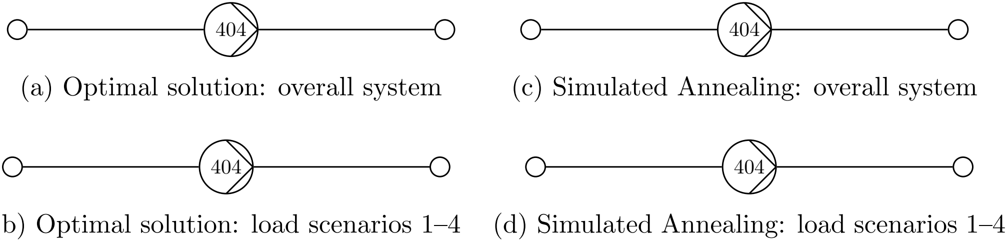
(c) Simulated Annealing: overall system

(b) Optimal solution: load scenarios 1–4

(d) Simulated Annealing: load scenarios 1–4

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B15_O_C_1

B15_O_C_2



(a) Optimal solution: overall system

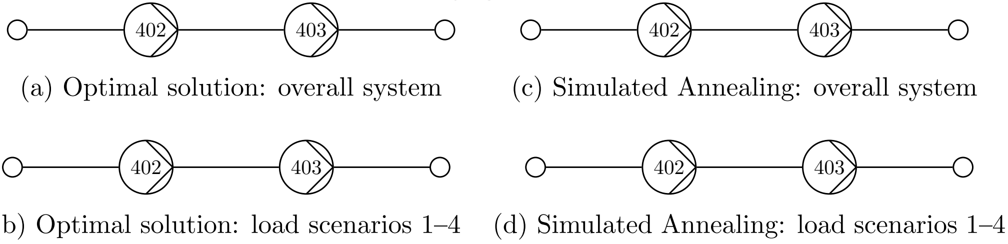
(c) Simulated Annealing: overall system

(b) Optimal solution: load scenarios 1–4

(d) Simulated Annealing: load scenarios 1–4

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B15_O_C_2

B15_O_D_1



(a) Optimal solution: overall system

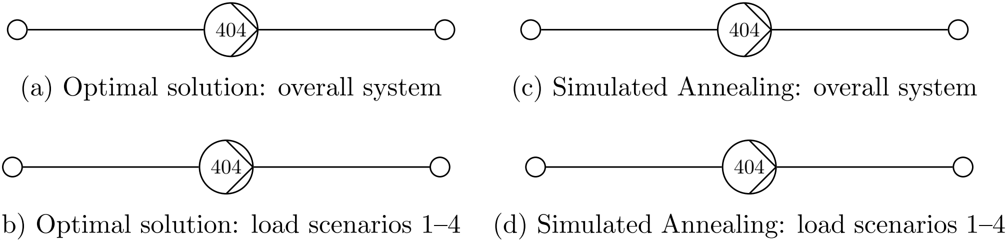
(c) Simulated Annealing: overall system

(b) Optimal solution: load scenarios 1–4

(d) Simulated Annealing: load scenarios 1–4

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B15_O_D_1

B15_O_D_2



(a) Optimal solution: overall system

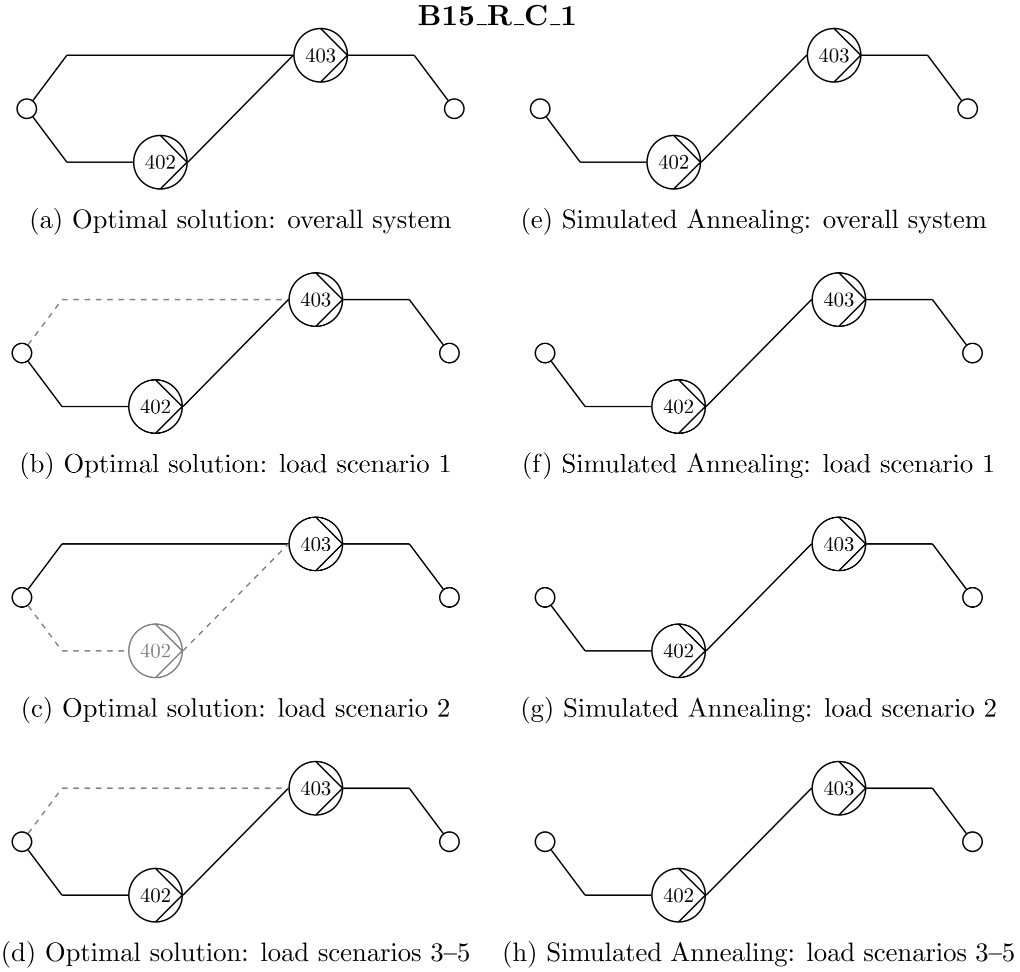
(c) Simulated Annealing: overall system

(b) Optimal solution: load scenarios 1–4

(d) Simulated Annealing: load scenarios 1–4

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B15_O_D_2

B15_R_C_1



(a) Optimal solution: overall system

(e) Simulated Annealing: overall system

(b) Optimal solution: load scenario 1

(f) Simulated Annealing: load scenario 1

(c) Optimal solution: load scenario 2

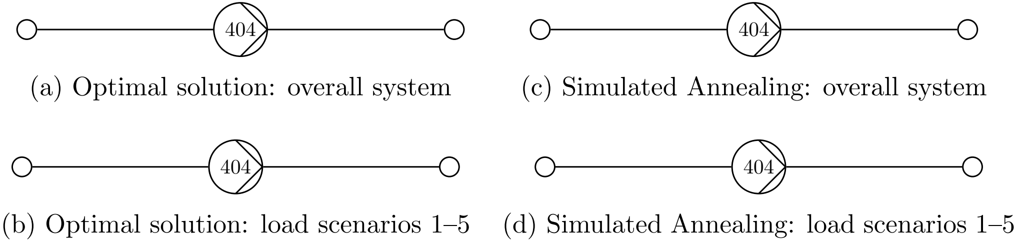
(g) Simulated Annealing: load scenario 2

(d) Optimal solution: load scenarios 3–5

(h) Simulated Annealing: load scenarios 3–5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B15_R_C_1

B15_R_C_2



(a) Optimal solution: overall system

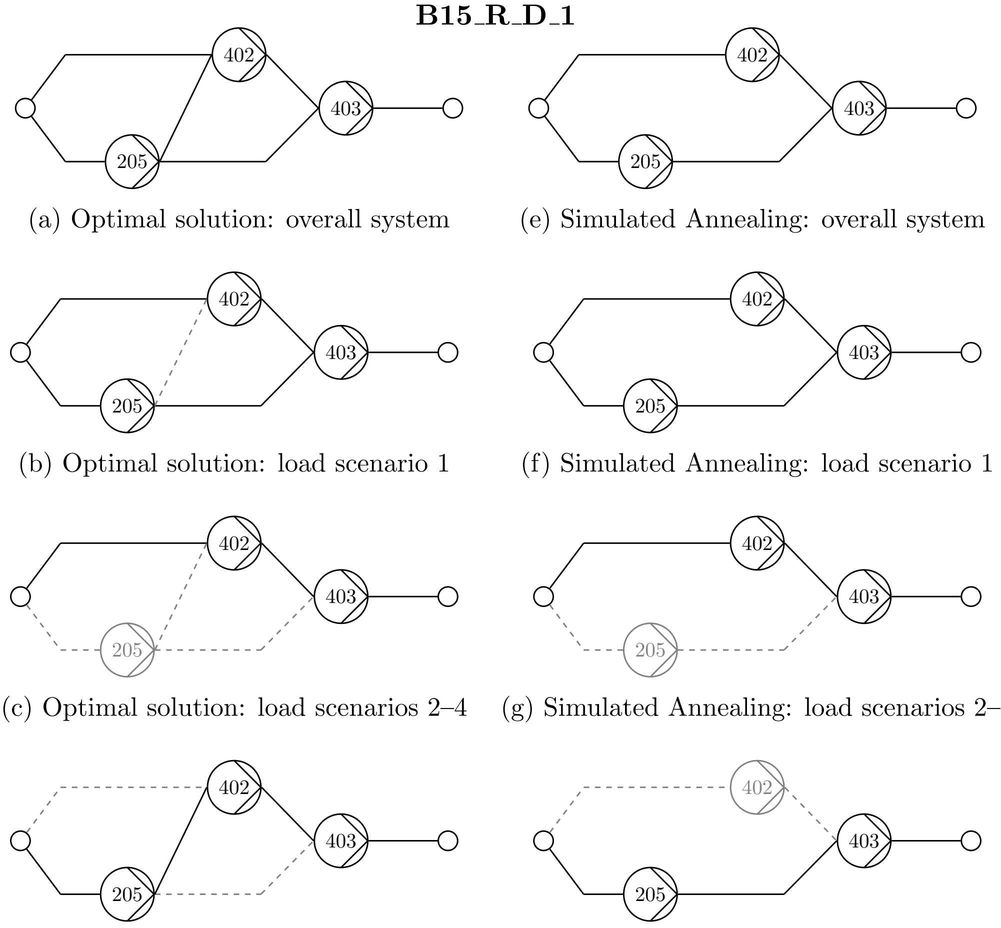
(c) Simulated Annealing: overall system

(b) Optimal solution: load scenarios 1–5

(d) Simulated Annealing: load scenarios 1–5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B15_R_C_2

B15_R_D_1



(a) Optimal solution: overall system (e) Simulated Annealing: overall system

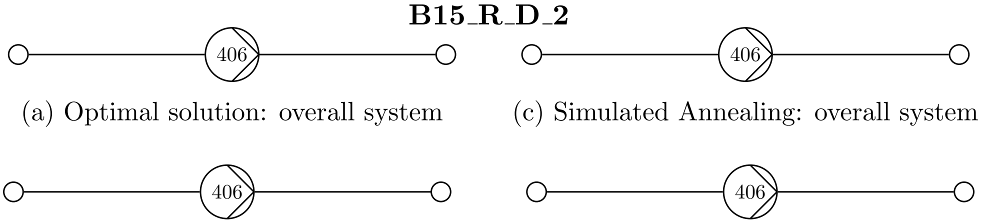
(b) Optimal solution: load scenario 1 (f) Simulated Annealing: load scenario 1

(c) Optimal solution: load scenarios 2–4 (g) Simulated Annealing: load scenarios 2–4

(d) Optimal solution: load scenario 5 (h) Simulated Annealing: load scenario 5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B15_R_D_1

B15_R_D_2

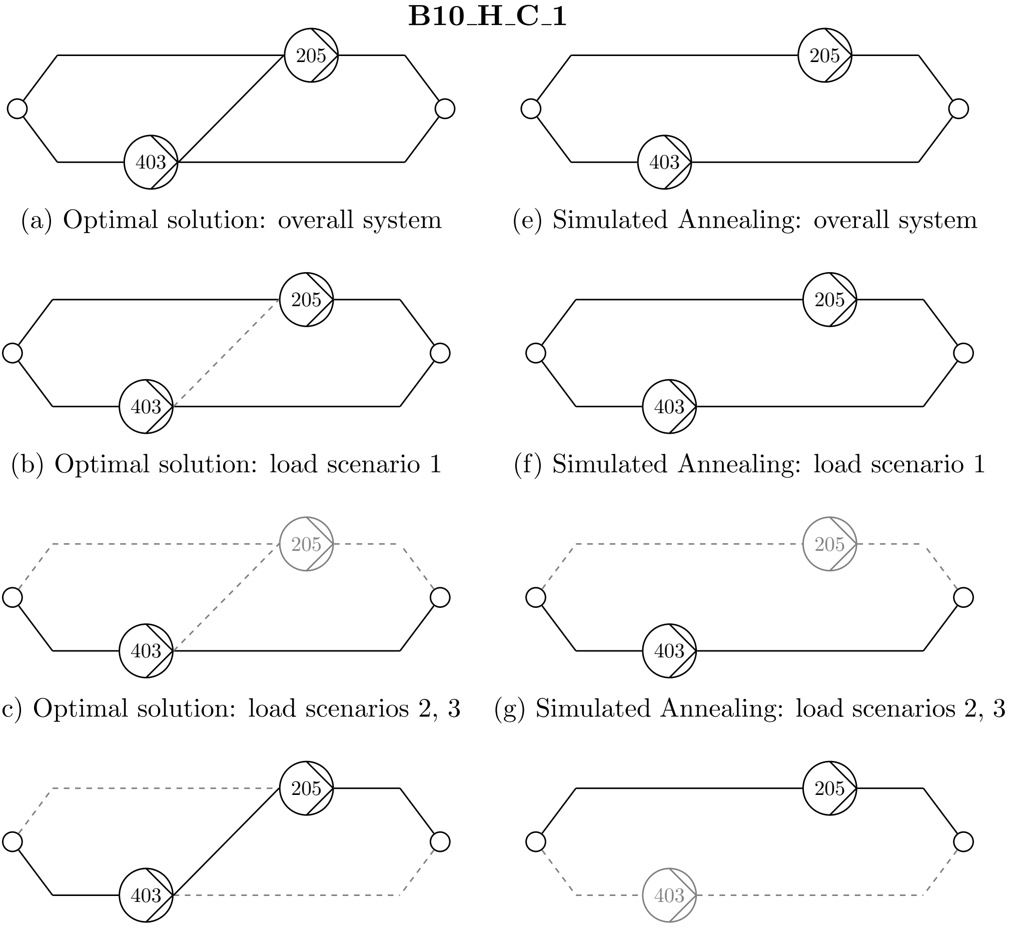


(a) Optimal solution: overall system (c) Simulated Annealing: overall system

(b) Optimal solution: load scenarios 1–5 (d) Simulated Annealing: load scenarios 1–5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B15_R_D_2

B10_H_C_1



(a) Optimal solution: overall system (e) Simulated Annealing: overall system

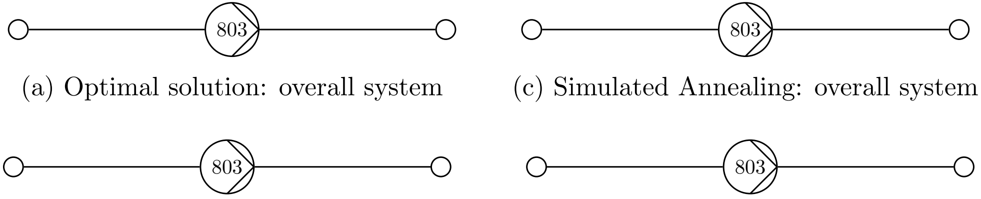
(b) Optimal solution: load scenario 1 (f) Simulated Annealing: load scenario 1

(c) Optimal solution: load scenarios 2, 3 (g) Simulated Annealing: load scenarios 2, 3

(d) Optimal solution: load scenarios 4, 5 (h) Simulated Annealing: load scenarios 4,5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B10_H_C_1

B10_H_C_2

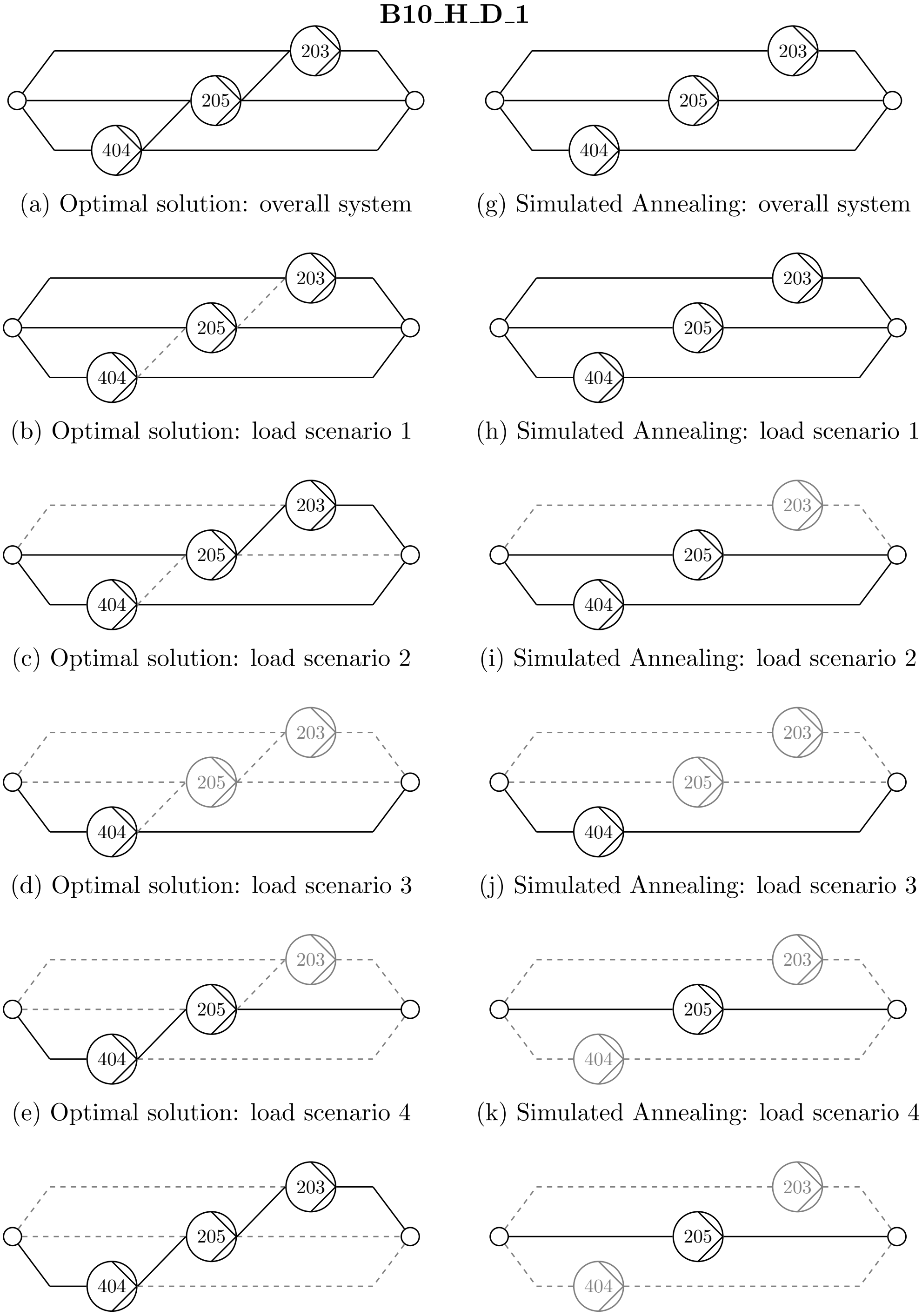


(a) Optimal solution: overall system (c) Simulated Annealing: overall system

(b) Optimal solution: load scenarios 1–5 (d) Simulated Annealing: load scenarios 1–5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B10_H_C_2

B10_H_D_1



(a) Optimal solution: overall system

(b) Optimal solution: load scenario 1

(c) Optimal solution: load scenario 2

(d) Optimal solution: load scenario 3

(e) Optimal solution: load scenario 4

(f) Optimal solution: load scenario 5

(g) Simulated Annealing: overall system

(h) Simulated Annealing: load scenario 1

(i) Simulated Annealing: load scenario 2

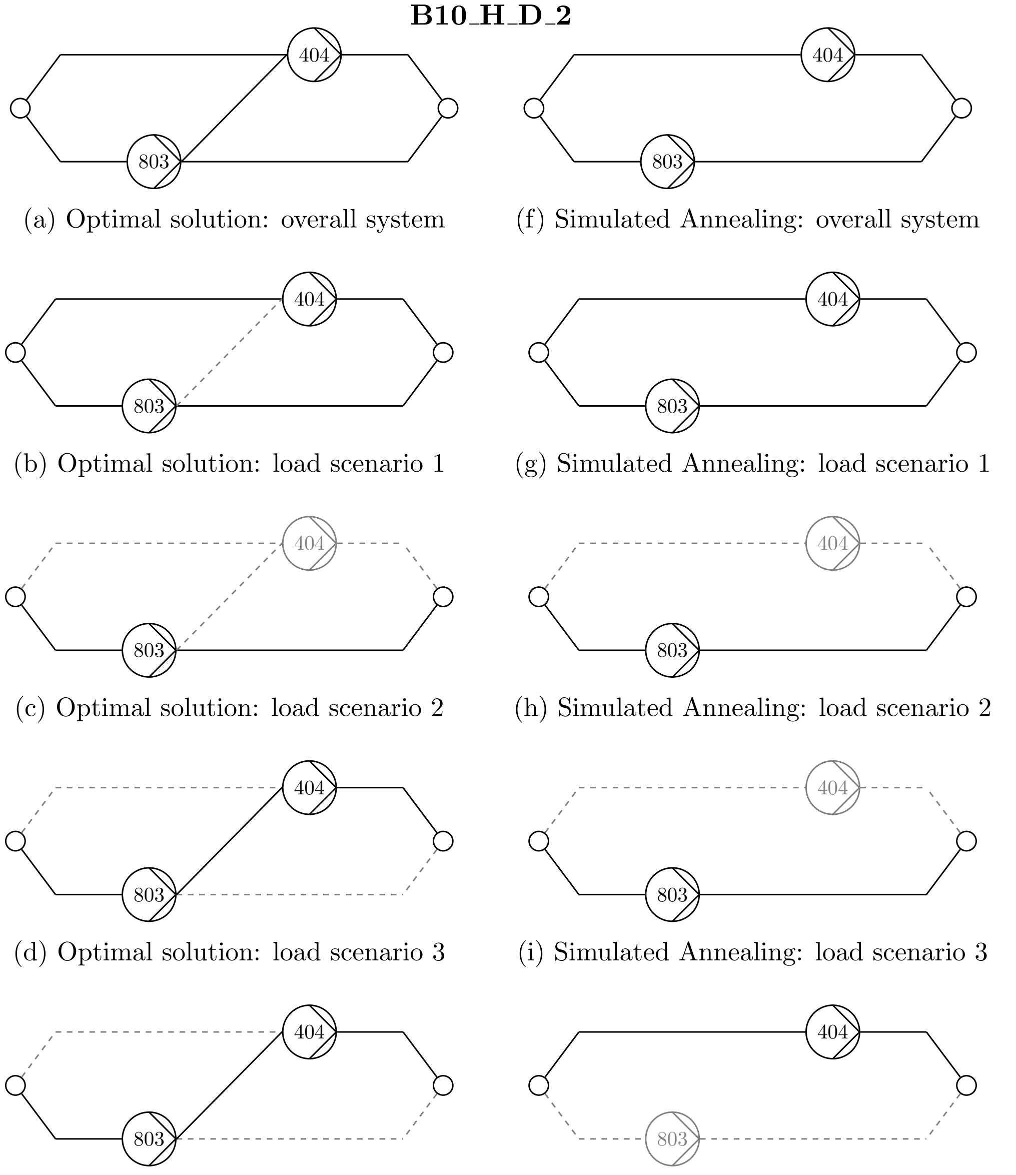
(j) Simulated Annealing: load scenario 3

(k) Simulated Annealing: load scenario 4

(l) Simulated Annealing: load scenario 5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B10_H_D_1

B10_H_D_2



(a) Optimal solution: overall system

(f) Simulated Annealing: overall system

(b) Optimal solution: load scenario 1

(g) Simulated Annealing: load scenario 1

(c) Optimal solution: load scenario 2

(h) Simulated Annealing: load scenario 2

(d) Optimal solution: load scenario 3

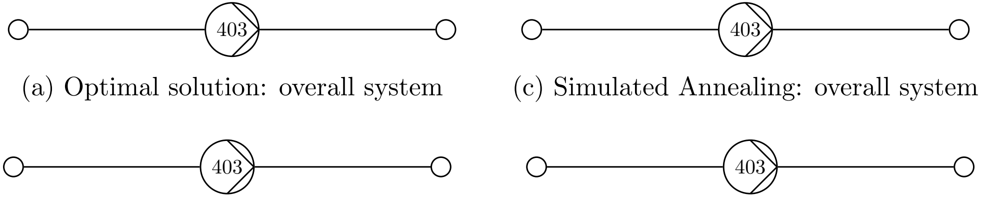
(i) Simulated Annealing: load scenario 3

(e) Optimal solution: load scenarios 4, 5

(j) Simulated Annealing: load scenarios 4, 5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B10_H_D_2

B10_O_C_1



(a) Optimal solution: overall system

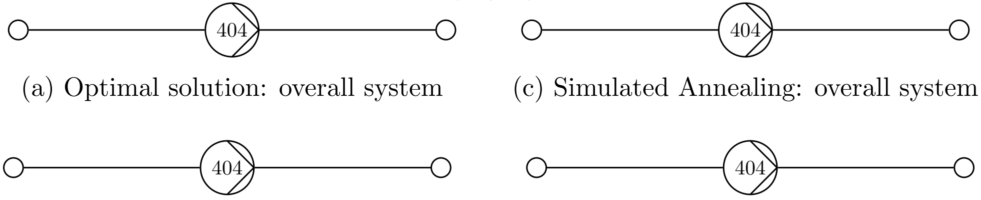
(c) Simulated Annealing: overall system

(b) Optimal solution: load scenarios 1–4

(d) Simulated Annealing: load scenarios 1–4

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B10_O_C_1

B10_O_C_2



(a) Optimal solution: overall system

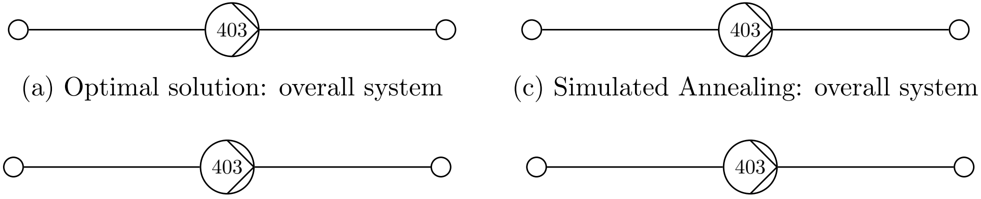
(c) Simulated Annealing: overall system

(b) Optimal solution: load scenarios 1–4

(d) Simulated Annealing: load scenarios 1–4

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B10_O_C_2

B10_O_D_1



(a) Optimal solution: overall system

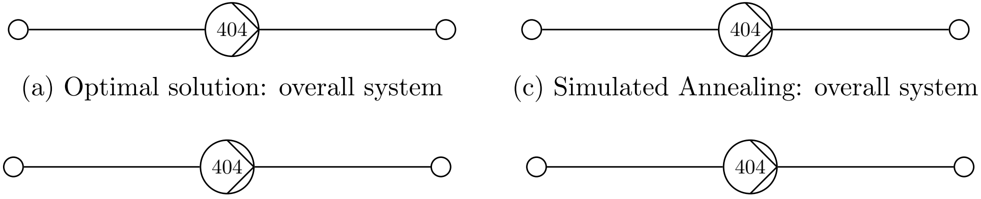
(c) Simulated Annealing: overall system

(b) Optimal solution: load scenarios 1–4

(d) Simulated Annealing: load scenarios 1–4

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B10_O_D_1

B10_O_D_2



(a) Optimal solution: overall system

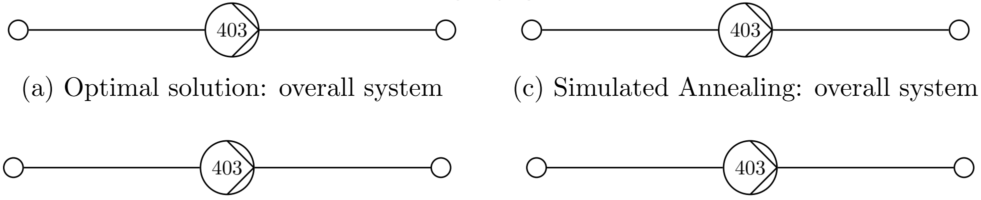
(c) Simulated Annealing: overall system

(b) Optimal solution: load scenarios 1–4

(d) Simulated Annealing: load scenarios 1–4

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B10_O_D_2

B10_R_C_1



(a) Optimal solution: overall system

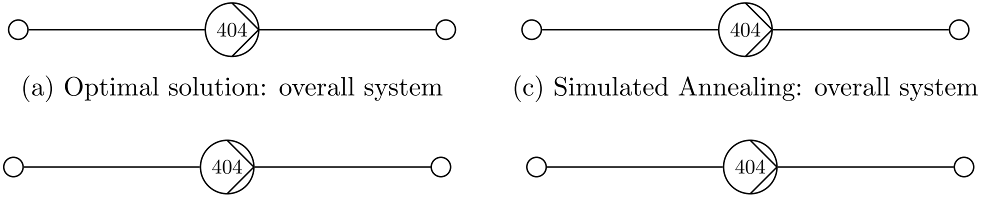
(c) Simulated Annealing: overall system

(b) Optimal solution: load scenarios 1–5

(d) Simulated Annealing: load scenarios 1–5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B10_R_C_1

B10_R_C_2



(a) Optimal solution: overall system

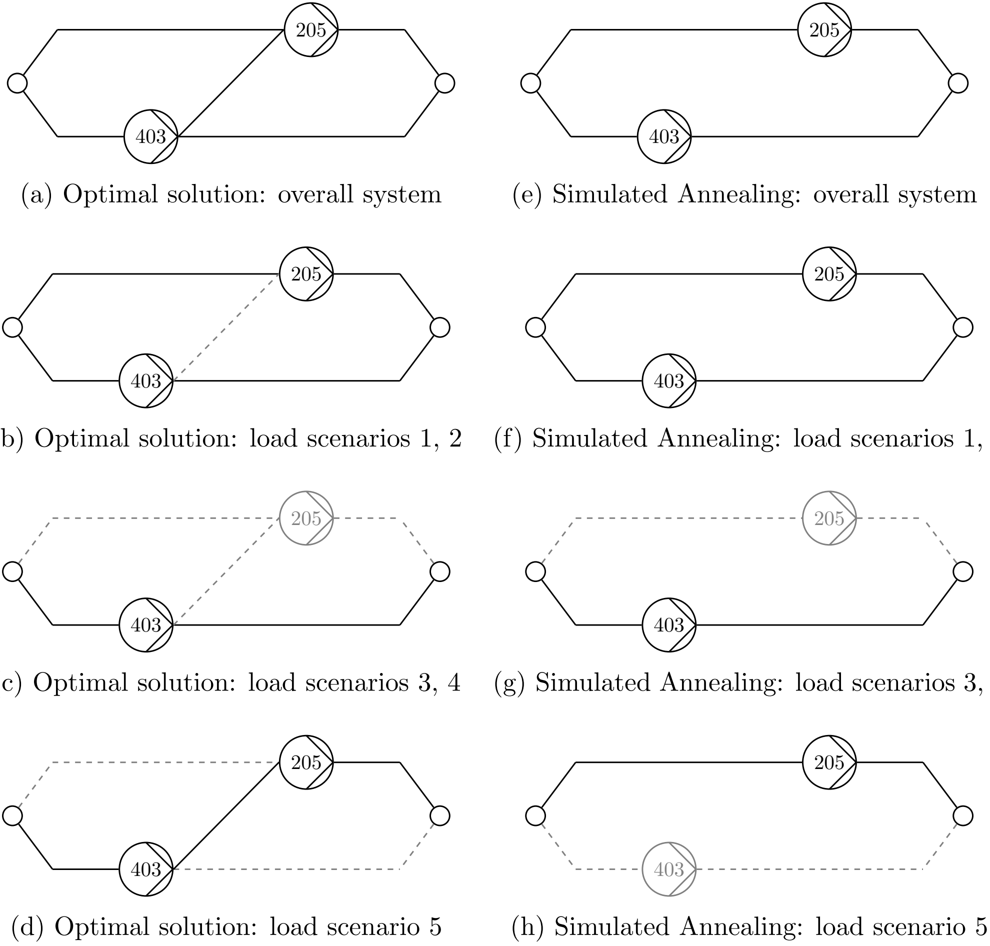
(c) Simulated Annealing: overall system

(b) Optimal solution: load scenarios 1–5

(d) Simulated Annealing: load scenarios 1–5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B10_R_C_2

B10_R_D_1



(a) Optimal solution: overall system

(e) Simulated Annealing: overall system

(b) Optimal solution: load scenarios 1, 2

(f) Simulated Annealing: load scenarios 1, 2

(c) Optimal solution: load scenarios 3, 4

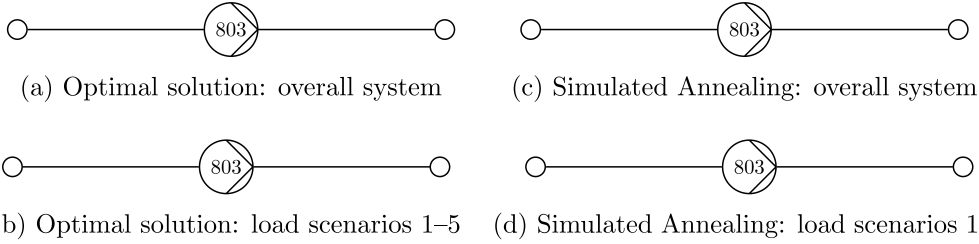
(g) Simulated Annealing: load scenarios 3, 4

(d) Optimal solution: load scenario 5

(h) Simulated Annealing: load scenario 5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B10_R_D_1

B10_R_D_2



(a) Optimal solution: overall system

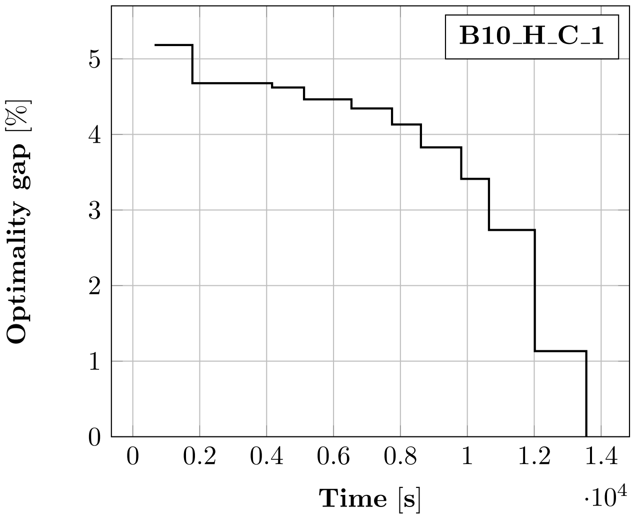
(c) Simulated Annealing: overall system

(b) Optimal solution: load scenarios 1–5

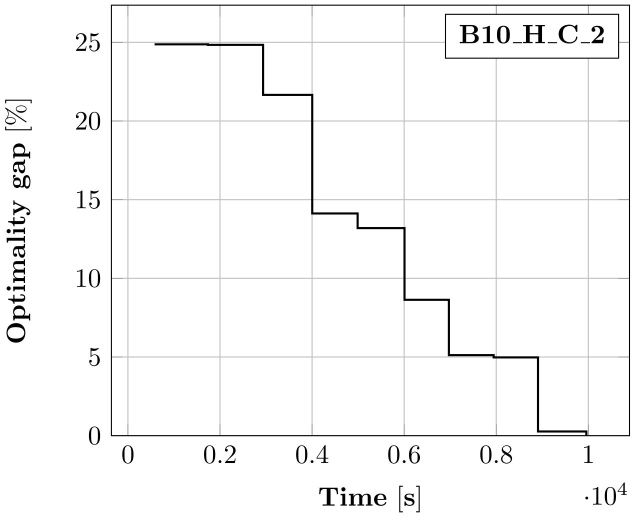
(d) Simulated Annealing: load scenarios 1–5

Comparison of the overall system and the operation in the individual load scenarios for the optimal solution and the solution found by Simulated Annealing for test instance B10_R_D_2

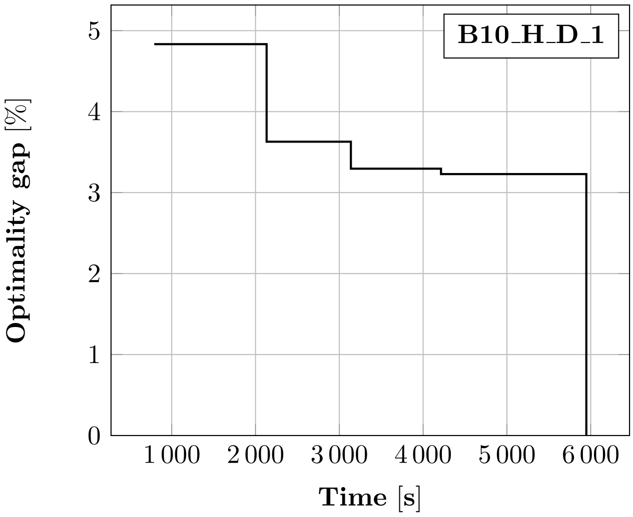
A.2 Overview of the Optimality Gap Progression for Section 5.5



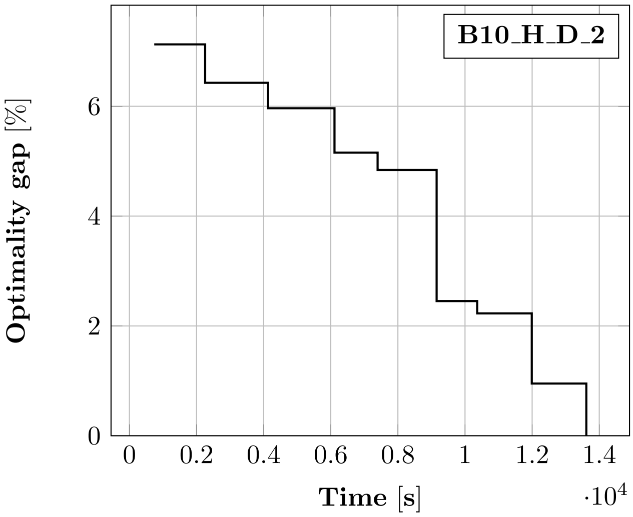
Branch-and-Bound: optimality gap progress over time for test instance B10_H_C_1



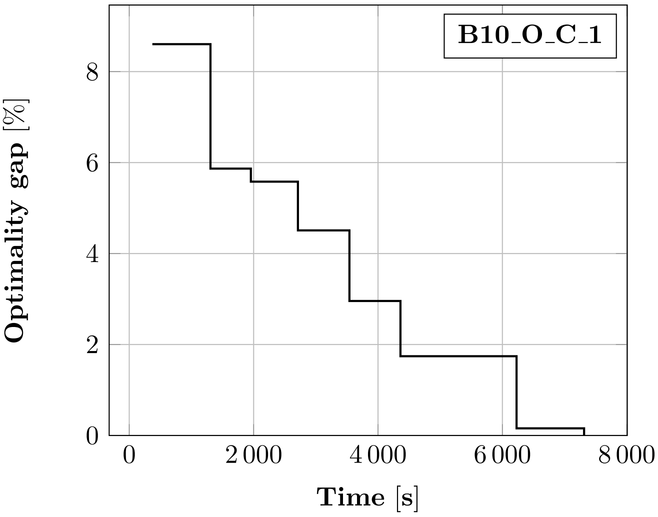
Branch-and-Bound: optimality gap progress over time for test instance B10_H_C_2



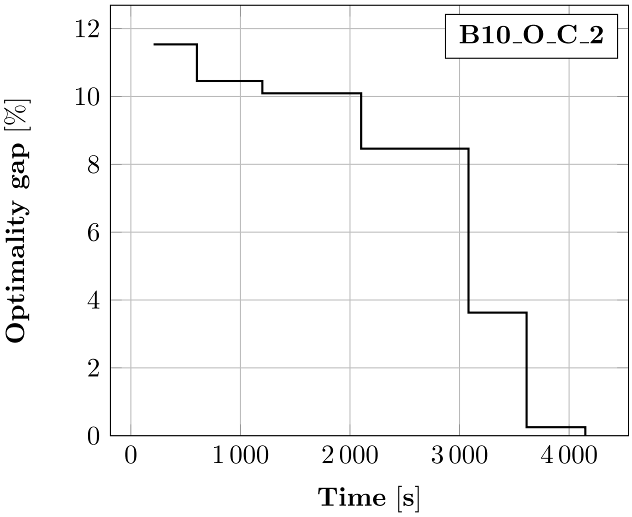
Branch-and-Bound: optimality gap progress over time for test instance B10_H_D_1



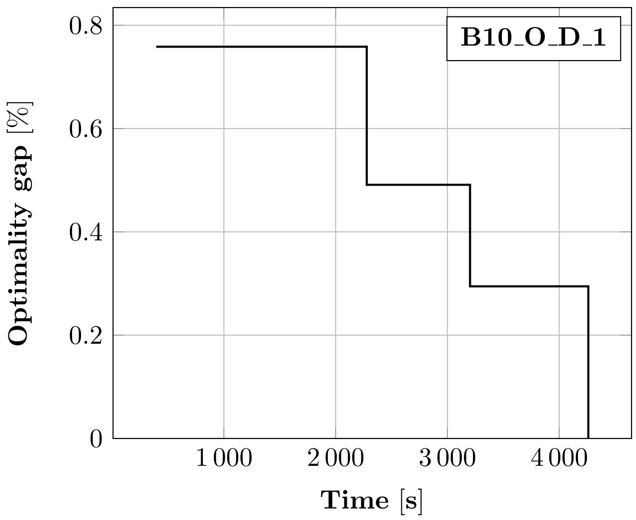
Branch-and-Bound: optimality gap progress over time for test instance B10_H_D_2



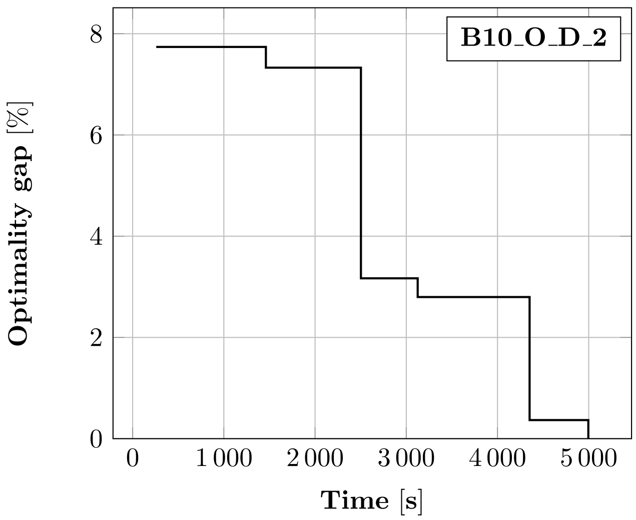
Branch-and-Bound: optimality gap progress over time for test instance B10_O_C_1



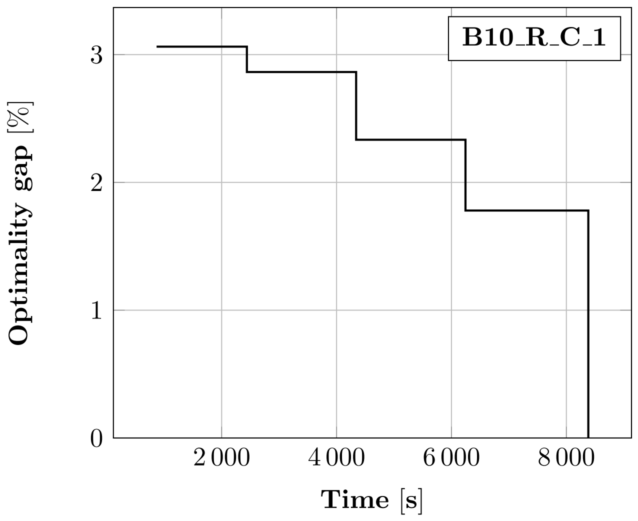
Branch-and-Bound: optimality gap progress over time for test instance B10_O_C_2



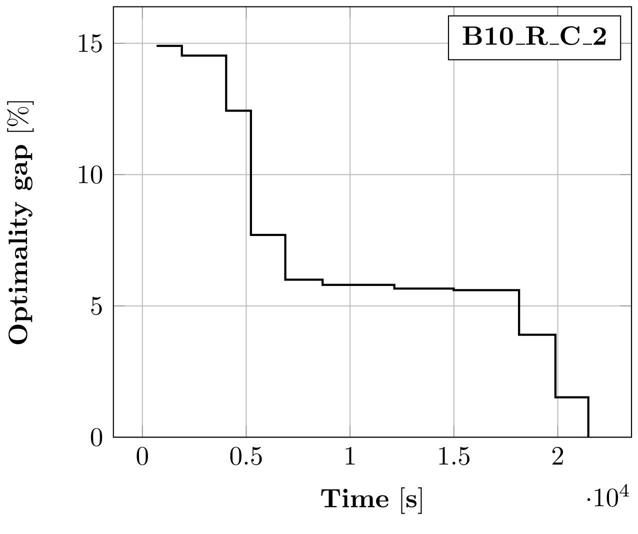
Branch-and-Bound: optimality gap progress over time for test instance B10_O_D_1



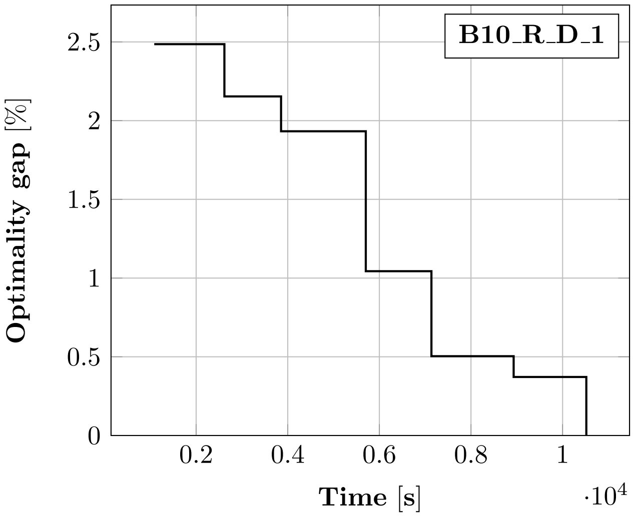
Branch-and-Bound: optimality gap progress over time for test instance B10_O_D_2



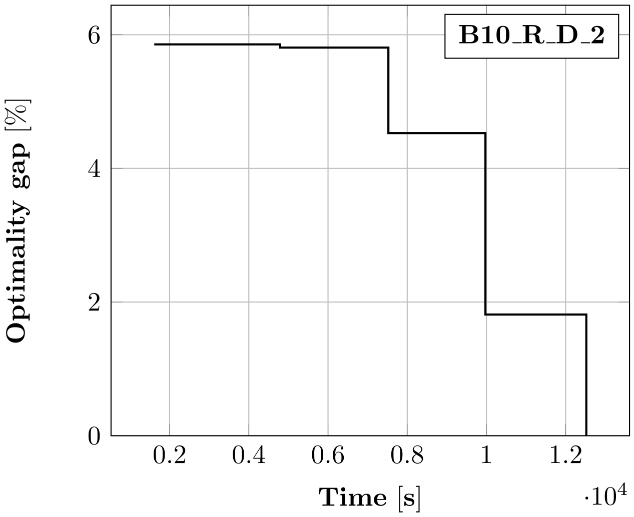
Branch-and-Bound: optimality gap progress over time for test instance B10_R_C_1



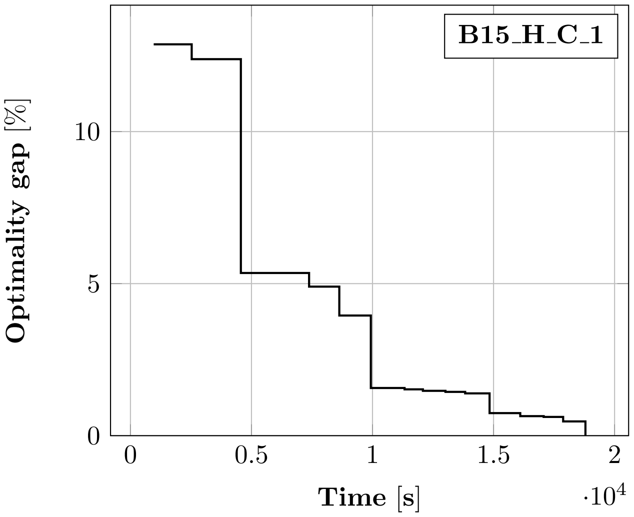
Branch-and-Bound: optimality gap progress over time for test instance B10_R_C_2



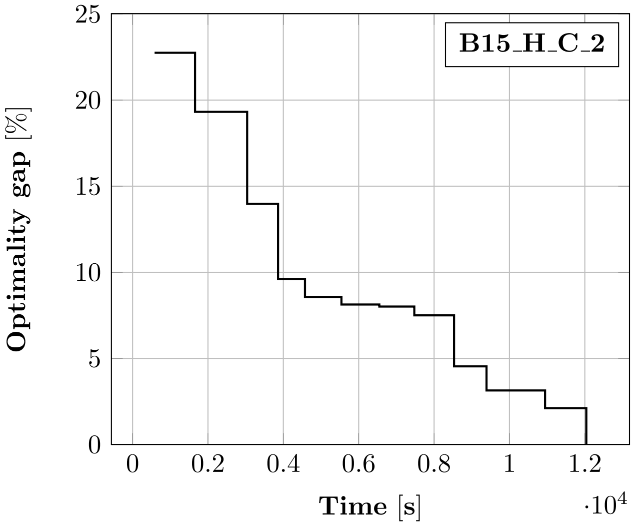
Branch-and-Bound: optimality gap progress over time for test instance B10_R_D_1



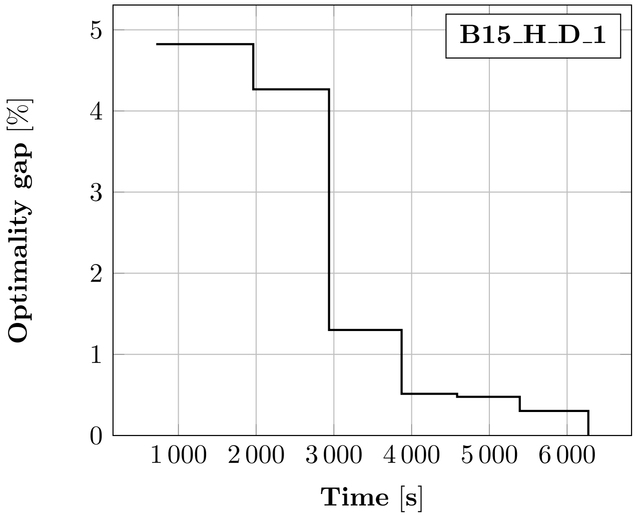
Branch-and-Bound: optimality gap progress over time for test instance B10_R_D_2



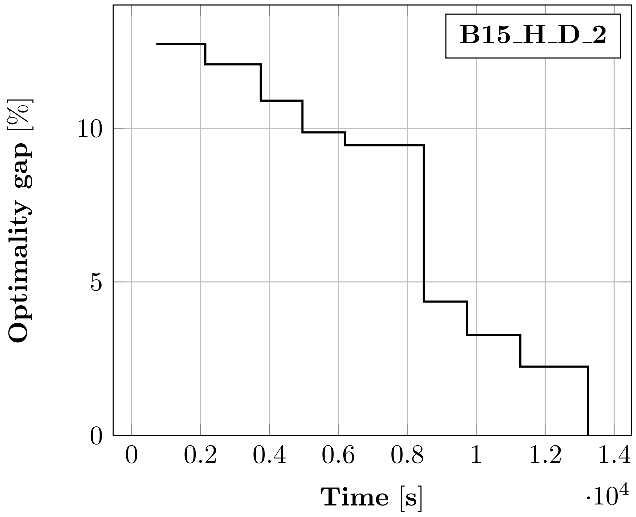
Branch-and-Bound: optimality gap progress over time for test instance B15_H_C_1



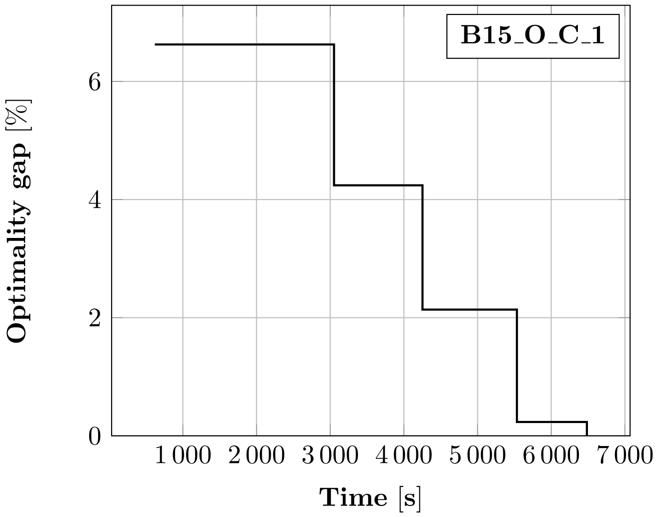
Branch-and-Bound: optimality gap progress over time for test instance B15_H_C_2



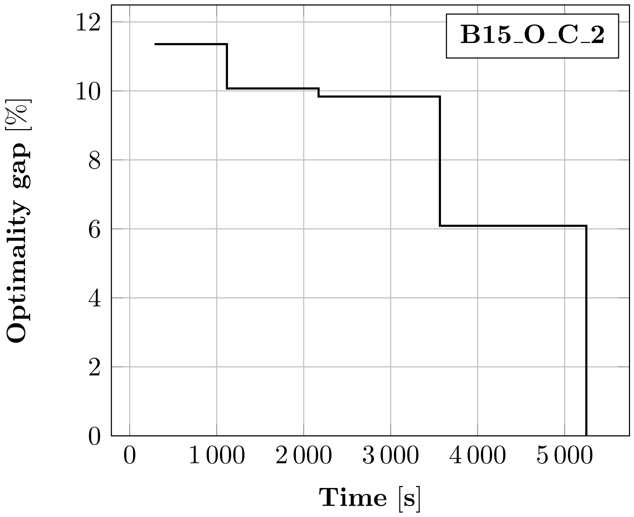
Branch-and-Bound: optimality gap progress over time for test instance B15_H_D_1



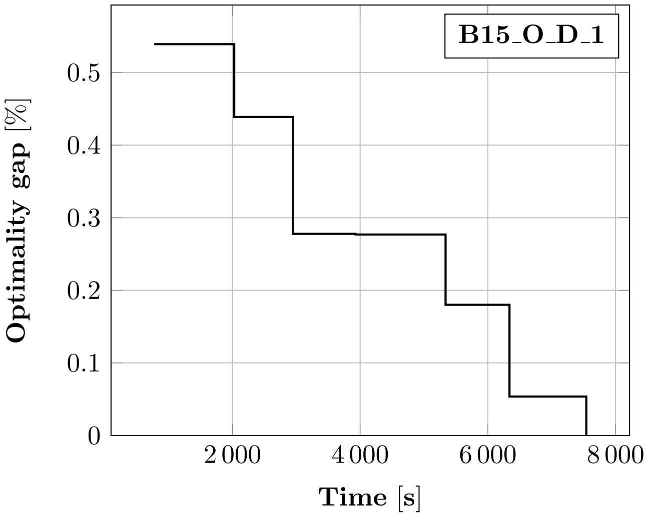
Branch-and-Bound: optimality gap progress over time for test instance B15_H_D_2



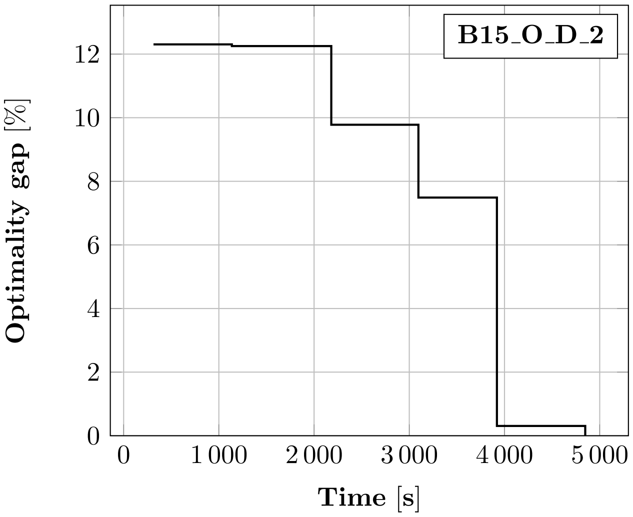
Branch-and-Bound: optimality gap progress over time for test instance B15_O_C_1



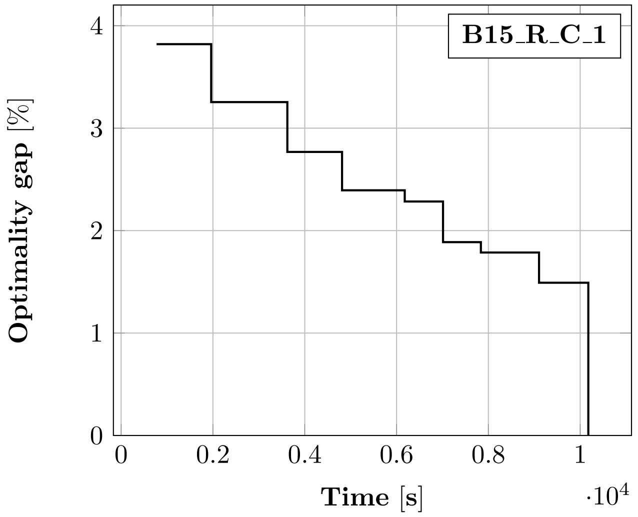
Branch-and-Bound: optimality gap progress over time for test instance B15_O_C_2



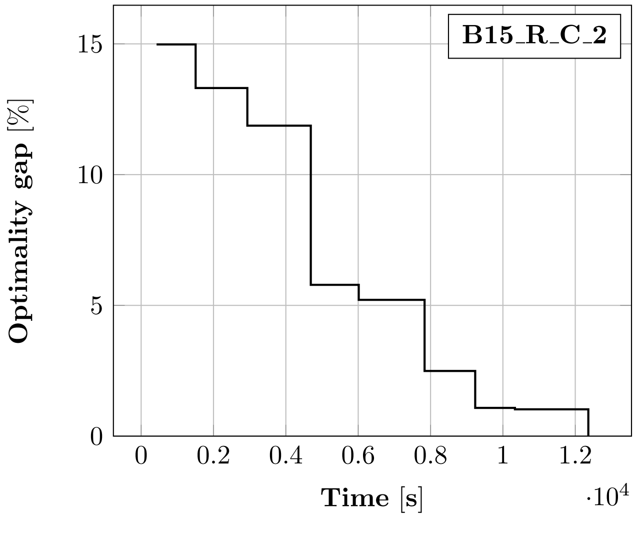
Branch-and-Bound: optimality gap progress over time for test instance B15_O_D_1



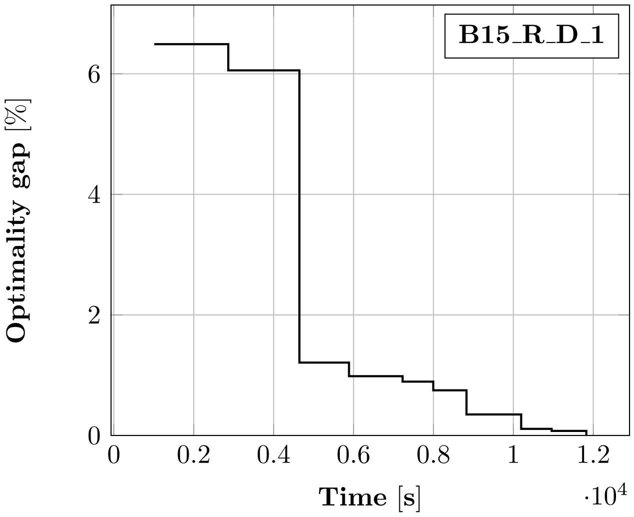
Branch-and-Bound: optimality gap progress over time for test instance B15_O_D_2



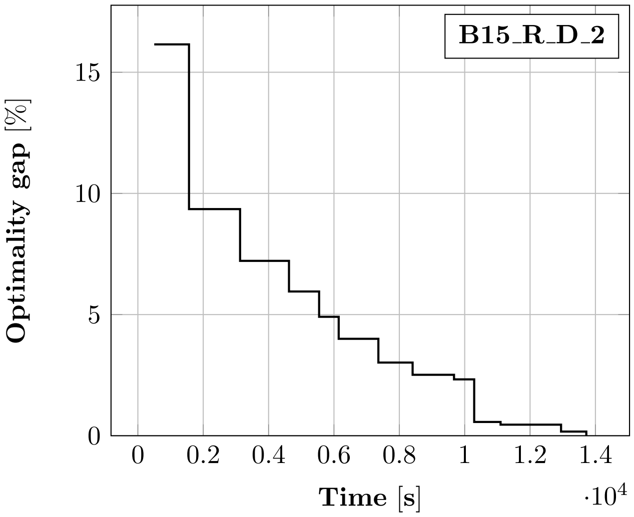
Branch-and-Bound: optimality gap progress over time for test instance B15_R_C_1



Branch-and-Bound: optimality gap progress over time for test instance B15_R_C_2



Branch-and-Bound: optimality gap progress over time for test instance B15_R_D_1



Branch-and-Bound: optimality gap progress over time for test instance B15_R_D_2

Bibliography

A. Afram and F. Janabi-Sharifi. Theory and applications of HVAC control systems—a review of model predictive control (MPC). *Build Environ*, 72:343–355, 2014.

L. C. Altherr. *Algorithmic system design under consideration of dynamic processes.* Dissertation, Technische Universität Darmstadt, 2016.

L. C. Altherr, T. Ederer, P. Pöttgen, U. Lorenz, and P. F. Pelz. Multicriterial optimization of technical systems considering multiple load and availability scenarios. In *Uncertainty in Mechanical Engineering II*, pages 247–256, Switzerland, 2015. Trans Tech Publications Ltd.

L. C. Altherr, T. Ederer, U. Lorenz, P. F. Pelz, and P. Pöttgen. Experimental validation of an enhanced system synthesis approach. In *Operations Research Proceedings 2014*, pages 1–7, Cham, 2016a. Springer.

L. C. Altherr, T. Ederer, U. Lorenz, P. F. Pelz, and P. Pöttgen. Designing a feedback control system via mixed-integer programming. In *Operations Research Proceedings 2014*, pages 121–127, Cham, 2016b. Springer.

L. C. Altherr, T. Ederer, L. S. Farnetane, A. Vergé, P. Pöttgen, and P. F. Pelz. Multicriterial design of a hydrostatic transmission system via mixed-integer programming. In *Operations Research Proceedings 2015*, pages 301–307, Cham, 2017a. Springer.

L. C. Altherr, T. Ederer, C. Schänzle, U. Lorenz, and P. F. Pelz. Algorithmic system design using scaling and affinity laws. In *Operations Research Proceedings 2015*, pages 605–611, Cham, 2017b. Springer.

L. C. Altherr, N. Brötz, I. Dietrich, T. Gally, F. Geßner, H. Kloberdanz, P. Leise, P. F. Pelz, P. D. Schlemmer, and A. Schmitt. Resilience in mechanical engineering—a concept for controlling uncertainty during design, production and usage phase of load-carrying structures. In *Uncertainty in Mechanical Engineering III*, pages 187–198, Switzerland, 2018a. Trans Tech Publications Ltd.

L. C. Altherr, B. Dörig, T. Ederer, P. F. Pelz, M. E. Pfetsch, and J. Wolf. A mixed-integer nonlinear program for the design of gearboxes. In *Operations Research Proceedings 2016*, pages 227–233, Cham, 2018b. Springer.

L. C. Altherr, P. Leise, M. E. Pfetsch, and A. Schmitt. Resilient layout, design and operation of energy-efficient water distribution networks for high-rise buildings using MINLP. *Optim Eng*, 20(2):605–645, 2019.

V. Andiappan. State-of-the-art review of mathematical optimisation approaches for synthesis of energy systems. *Process Integr Optim Sustain*, 1(3):165–188, 2017.

S. Artina and J. Walker. Sull'uso della programmazione a valori misti nel dimensionamento di costo minimo di reti in pressione [Using mixed-integer programming for cost-minimal sizing of pressurized networks]. *Atti dell'Accademia delle Scienze dell'Istituto di Bologna*, 271, 1983.

A. Ashouri, S. S. Fux, M. J. Benz, and L. Guzzella. Optimal design and operation of building services using mixed-integer linear programming techniques. *Energy*, 59:365–376, 2013.

B. Baeten, F. Rogiers, and L. Helsen. Energy cost reduction by optimal control of ideal sensible thermal energy storage in domestic space heating. In *Proceedings of the Eurotherm Seminar 99 conference—Advances in thermal energy storage*, pages 01–045, Stockholm, 2014.

M. Baglione. Building sustainability into control systems, 2011. https://engfac.cooper.edu/melody/411 (accessed May 07, 2021).

B. Bahl. *Optimization-Based Synthesis of Large-Scale Energy Systems by Time-Series Aggregation*. Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, 2018.

B. Bahl, S. Goderbauer, F. Arnold, P. Voll, M. Lübbecke, A. Bardow, and A. M. C. A. Koster. DESSLib—benchmark instance structure, component models and parameters. Technical report, Rheinisch-Westfälische Technische Hochschule Aachen, 2016.

B. Bahl, M. Hennen, M. Lampe, P. Voll, and A. Bardow. Optimization-based synthesis of resource-efficient utility systems. In *Resource Efficiency of Processing Plants: Monitoring and Improvement*, pages 373–401. Wiley Online Library, 2018.

A. Ben-Tal, L. El Ghaoui, and A. S. Nemirovski. *Robust Optimization*. Princeton University Press, Princeton, 2009.

P. Betz. *Über die Anwendung exakter Optimierungsverfahren zur Planung von Druckerhöhungsanlagen [On the use of exact optimization methods for planning booster stations]*. Dissertation, Technische Universität Darmstadt, 2017.

J. R. Birge and F. Louveaux. *Introduction to Stochastic Programming*. Springer, New York, 2nd edition, 2011.

G. Bonvin, S. Demassey, C. Le Pape, N. Maïzi, V. Mazauric, and A. Samperio. A convex mathematical program for pump scheduling in a class of branched water networks. *Appl Energy*, 185:1702–1711, 2017.

H. Booth and R. Tarjan. Finding the minimum-cost maximum flow in a series-parallel network. *J Algorithms*, 15(3):416–446, 1993.

I. Boussaïd, J. Lepagnot, and P. Siarry. A survey on optimization metaheuristics. *Inf Sci*, 237:82–117, 2013.

C. Bragalli, C. D'Ambrosio, J. Lee, A. Lodi, and P. Toth. On the optimal design of water distribution networks: A practical MINLP approach. *Optim Eng*, 13(2): 219–246, 2012.

J. Burgschweiger, B. Gnädig, and M. C. Steinbach. Nonlinear programming techniques for operative planning in large drinking water networks. Technical Report ZIB-Report 05-31, Zuse Institute Berlin, 2005.

J. Burgschweiger, B. Gnädig, and M. C. Steinbach. Optimization models for operative planning in drinking water networks. *Optim Eng*, 10:43–73, 2009.

X. Cai, D. C. McKinney, and L. S. Lasdon. Solving nonlinear water management models using a combined genetic algorithm and linear programming approach. *Adv Water Resour*, 24(6):667–676, 2001.

Y.-C. Chang. A novel energy conservation method—optimal chiller loading. *Electr Power Syst Res*, 69(2):221–226, 2004.

J. Clausen. Branch and bound algorithms—principles and examples. Technical report, Department of Computer Science, University of Copenhagen, 1999.

A. L. H. Costa, J. L. de Medeiros, and F. L. P. Pessoa. Optimization of pipe networks including pumps by simulated annealing. *Br J Chem Eng*, 17(4-7): 887–896, 2000.

M. C. Cunha and L. Ribeiro. Tabu search algorithms for water network optimization. *Eur J Oper Res*, 157(3):746–758, 2004.

M. C. Cunha and J. Sousa. Water distribution network design optimization: Simulated annealing approach. *J Water Resour Plan Manag*, 125(4):215–221, 1999.

C. D'Ambrosio, A. Lodi, S. Wiese, and C. Bragalli. Mathematical programming techniques in water network optimization. *Eur J Oper Res*, 243(3):774–788, 2015.

G. C. Dandy, A. R. Simpson, and L. J. Murphy. Optimum design and operation of pumped water distribution systems. In *International Conference on hydraulics in civil engineering: "Hydraulics Working with the Environment"*, pages 149–155, Brisbane, 1994. Institution of Engineers.

A. De Corte and K. Sörensen. Optimisation of gravity-fed water distribution network design: A critical review. *Eur J Oper Res*, 228(1):1–10, 2013.

B. Delaunay. Sur la sphère vide [On the empty sphere]. *Bull Acad Sci USSR*, 6: 793–800, 1934.

K. Deng, Y. Sun, A. Chakraborty, Y. Lu, J. Brouwer, and P. G. Mehta. Optimal scheduling of chiller plant with thermal energy storage using mixed integer linear programming. In *2013 American Control Conference*, pages 2958–2963, Washington, D.C., 2013. IEEE.

K. Deng, Y. Sun, S. Li, Y. Lu, J. Brouwer, P. G. Mehta, M. Zhou, and A. Chakraborty. Model predictive control of central chiller plant with thermal energy storage via dynamic programming and mixed-integer linear programming. *IEEE Trans Autom Sci Eng*, 12(2):565–579, 2015.

DIN 1988-500. Codes of practice for drinking water installations—Part 500: Pressure boosting stations with RPM-regulated pumps; DVGW code of practice, Feb. 2011.

P. Domschke, B. Geißler, O. Kolb, J. Lang, A. Martin, and A. Morsi. Combination of nonlinear and linear optimization of transient gas networks. *INFORMS J Comput*, 23(4):605–617, 2011.

P. M. Doran. *Bioprocess engineering principles.* Academic Press, Waltham, 2nd edition, 2011.

B. Dörig, T. Ederer, P. Hedrich, U. Lorenz, P. F. Pelz, and P. Pöttgen. Technical Operations Research (TOR) exemplified by a hydrostatic power transmission system. In *9. IFK—9th International Fluid Power Conference*, Aachen, 2014.

B. Dörig, T. Ederer, P. F. Pelz, M. E. Pfetsch, and J. Wolf. Gearbox design via mixedinteger programming. In *ECCOMAS Congress 2016—VII European Congress on Computational Methods in Applied Sciences and Engineering*, Crete Island, 2016.

T. Ederer, U. Lorenz, M. Metzler, P. F. Pelz, and P. Pöttgen. Global system optimization and scaling for turbo systems and machines. In *ISROMAC-15—15th International Symposium on Transport Phenomena and Dynamics of Rotating Machinery*, Honolulu, 2014.

G. Eiger, U. Shamir, and A. Ben-Tal. Optimal design of water distribution networks. *Water Resour Res*, 30:2637–2646, 1994.

H. Elmqvist, H. Tummescheit, and M. Otter. Object-oriented modeling of thermofluid systems. In *Proceedings of the 3rd International Modelica Conference*, pages 269–286, Linköping, 2003.

European Commission. An EU strategy on heating and cooling, 2016. https://ec.europa.eu/energy/sites/ener/files/documents/1_EN_ACT_part1_v14.pdf (accessed May 07, 2021).

European Commission. Ecodesign pump review—study of Commission Regulation (EU) No. 547/2012 (Ecodesign requirements for water pumps) – Extended report (final version), 2018. https://www.eceee.org/static/media/uploads/site-2/ecodesign/products/Electric%20pumps%20ENER%20Lot%2011/review_study_on_water_pumps_-_december_2018.pdf (accessed May 07, 2021).

European Statistical Office (Eurostat). Gross and net production of electricity and derived heat by type of plant and operator [nrg_ind_peh], 2020. https://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=nrg_ind_peh&lang=en (accessed May 07, 2021).

Federal Ministry for Economic Affairs and Energy (BMWi). Energy efficiency — Made in Germany: Energy efficiency in industry, building service technology and transport, 2010. https://www.bmwi.de/Redaktion/EN/Publikationen/2010-energy-efficiency-made-in-germany.pdf?__blob=publicationFile&v=1 (accessed May 07, 2021).

Federal Ministry for Economic Affairs and Energy (BMWi). Energy efficiency: "Germany makes it efficient", 2019. https://www.bmwi.de/Redaktion/EN/Dossier/energy-efficiency.html (accessed May 07, 2021).

C. A. Floudas and X. Lin. Mixed integer linear programming in process scheduling: Modeling, algorithms, and applications. *Ann Oper Res*, 131(1):131–162, 2005.

C. Frangopoulos, M. von Spakovsky, and E. Sciubba. A brief review of methods for the design and synthesis optimization of energy systems. *Int J Thermodyn*, 5 (4):151–160, 2002.

A. Fügenschuh, H. Homfeld, H. Schülldorf, and S. Vigerske. Mixed-integer nonlinear problems in transportation applications. In *EngOpt 2010 Proceedings of the 2nd International Conference on Engineering Optimization*, 2010.

A. Fügenschuh, R. Gollmer, B. Geißler, C. Hayn, R. Henrion, B. Hiller, J. Humpola, T. Koch, T. Lehmann, A. Martin, R. Mirkov, A. Morsi, J. Rövekamp, L. Schewe, M. Schmidt, R. Schultz, R. Schwarz, J. Schweiger, C. Stangl, and B. Willert. Mathematical optimization for challenging network planning problems in unbundled liberalized gas markets. Technical Report ZIB-Report 13-13, Zuse Institute Berlin, 2013.

A. Fügenschuh, C. Hayn, and D. Michaels. Mixed-integer linear methods for layout-optimization of screening systems in recovered paper production. *Optim Eng*, 15 (2):533–573, 2014.

B. Geißler. *Towards Globally Optimal Solutions for MINLPs by Discretization Techniques with Applications in Gas Network Optimization.* Dissertation, Universität Erlangen-Nürnberg, 2011.

B. Geißler, O. Kolb, J. Lang, G. Leugering, A. Martin, and A. Morsi. Mixed integer linear models for the optimization of dynamical transport networks. *Math Methods Oper Res*, 73(3):339–362, 2011.

B. Geißler, A. Martin, A. Morsi, and L. Schewe. Using piecewise linear functions for solving MINLPs. In *Mixed Integer Nonlinear Programming*, volume 154 of *The IMA Volumes in Mathematics and its Applications*, pages 287–314. Springer, New York, 2012.

German Engineering Association (VDMA). A pump is not a bulb, 2019. https://pu.vdma.org/documents/105981/45703829/2019-11-28%20Article%20EPA%20engl_1574932601958.pdf/e54ecbec-7905-1082-33a1-cf886bb3dad9 (accessed May 07, 2021).

A. M. Gleixner, H. Held, W. Huang, and S. Vigerske. Towards globally optimal operation of water supply networks. *Numer Algebra Control Optim*, 2(4):695–711, 2012.

T. F. Groß, P. Pöttgen, and P. F. Pelz. Analytical approach for the optimal operation of pumps in booster systems. *J Water Resour Plan Manag*, 143(8):04017029, 2017.

M. Guignard and S. Kim. Lagrangean decomposition: A model yielding stronger lagrangean bounds. *Math Program*, 39(2):215–228, 1987.

M. Haller, W. Streicher, E. Andersen, and S. Furbo. Energy-efficient design of a water supply system for skyscrapers by mixed-integer nonlinear programming. In *EffStock 2009—The 11th Intl Conference on Thermal Energy Storage for Efficiency and Sustainability*, pages 1–9, Stockholm, 2009.

M. Hartisch. *Quantified integer programming with polyhedral and decision-dependent uncertainty.* Dissertation, Universität Siegen, 2020.

M. Hartisch, A. Herbst, U. Lorenz, and J. B. Weber. Towards resilient process networks—designing booster stations via quantified programming. In *Uncertainty in Mechanical Engineering III*, pages 199–210, Switzerland, 2018. Trans Tech Publications Ltd.

S. S. Hashemi, M. Tabesh, and B. Ataeekia. Ant-colony optimization of pumping schedule to minimize the energy cost using variable-speed pumps in water distribution networks. *Urban Water J*, 11(5):335–347, 2014.

M. Herrera, E. Abraham, and I. Stoianov. A graph-theoretic framework for assessing the resilience of sectorised water distribution networks. *Water Resour Manage*, 30(5):1685–1699, 2016.

F. P. Incropera, D. P. DeWitt, T. L. Bergman, and A. S. Lavine. *Fundamentals of Heat and Mass Transfer*. John Wiley & Sons, Hoboken, 6th edition, 2007.

International Renewable Energy Agency (IRENA). *Global energy transformation: A roadmap to 2050 (2019 edition)*. International Renewable Energy Agency, Abu Dhabi, 2019.

ISO/ASME 14414. Pump system energy assessment; ISO code of practice, Sept. 2019.

K. Kapoor. *Modeling and Optimization for Energy Efficient Large Scale Cooling Operation*. Dissertation, University of Texas at Austin, 2013.

T. Kashima and S. P. Boyd. Cost optimal operation of thermal energy storage system with real-time prices. In *Proceedings of the 2013 International Conference on Control, Automation and Information Sciences*, pages 233–237, Nha Trang, 2013.

T.-S. Lee, K.-Y. Liao, and W.-C. Lu. Evaluation of the suitability of empirically-based models for predicting energy performance of centrifugal water chillers with variable chilled water flow. *Appl Energy*, 93:583–595, 2012.

P. Leise and L. C. Altherr. Optimizing the design and control of decentralized water supply systems—a case-study of a hotel building. In *EngOpt 2018 Proceedings of the 6th International Conference on Engineering Optimization*, pages 1241–1252, Cham, 2019. Springer.

P. Leise, L. C. Altherr, and P. F. Pelz. Energy-efficient design of a water supply system for skyscrapers by mixed-integer nonlinear programming. In *Operations Research Proceedings 2017*, pages 475–481, Cham, 2018. Springer.

P. Leise, L. C. Altherr, and P. F. Pelz. Technical operations research (TOR)—algorithms, not engineers, design optimal energy efficient and resilient cooling systems. In *FAN 2018—Proceedings of the International Conference on Fan Noise, Aerodynamics, Applications and Systems*, Darmstadt, 2019a.

P. Leise, L. C. Altherr, N. Simon, and P. F. Pelz. Finding global-optimal gearbox designs for battery electric vehicles. In H. A. Le Thi, H. M. Le, and T. Pham Dinh, editors, *WCGO19—Optimization of Complex Systems: Theory, Models, Algorithms and Applications*, pages 916–925, Cham, 2019b. Springer.

R. Lejano. Optimizing the layout and design of branched pipeline water distribution systems. *Irrig Drain Syst*, 20(1):125–137, 2006.

S. Lingireddy and D. Wood. Improved operation of water distribution systems using variable-speed pumps. *J Energy Eng*, 124(3):90–103, 1998.

I.-S. Lorenz and P. F. Pelz. Optimal resilience enhancement of water distribution systems. *Water*, 12(9):2602, 2020.

Y. Ma, F. Borrelli, B. Hencey, A. Packard, and S. Bortoff. Model predictive control of thermal energy storage in building cooling systems. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pages 392–397, New York, 2009. IEEE.

P. A. MacMahon. Yoke-chains and multipartite compositions in connexion with the analytical forms called "trees". *Proc Lond Math Soc*, 22:330–346, 1890.

D. Mahlke, A. Martin, and S. Moritz. A mixed integer approach for time-dependent gas network optimization. *Optim Methods Softw*, 25(4):625–644, 2010.

H. R. Maier, A. R. Simpson, A. C. Zecchin, W. K. Foong, K. Y. Phang, H. Y. Seah, and C. L. Tan. Ant colony optimization for design of water distribution systems. *J Water Resour Plan Manag*, 129(3):200–209, 2003.

H. Mala-Jetmarova, N. Sultanova, and D. Savić. Lost in optimisation of water distribution systems? A literature review of system operation. *Environ Model Softw*, 93:209–254, 2017.

H. Mala-Jetmarova, N. Sultanova, and D. Savić. Lost in optimisation of water distribution systems? A literature review of system design. *Water*, 10(3):307, 2018.

A. Marchi, G. Dandy, A. Wilkins, and H. Rohrlach. Methodology for comparing evolutionary algorithms for optimization of water distribution systems. *J Water Resour Plan Manag*, 140:22–31, 2014.

L. R. Marinho, M. das Chagas Moura, B. S. da Cunha, and I. D. Lins. Optimization of investments in the resilience of water distribution systems subject to interruptions. *Water Resour Manage*, 34:929–954, 2020.

A. Martin, M. Möller, and S. Moritz. ixed integer models for the stationary case of gas network optimization. *Math Program*, 105:563–582, 2006.

A. Martin, K. Klamroth, J. Lang, G. Leugering, A. Morsi, M. Oberlack, M. Ostrowski, and R. Rosen. *Mathematical Optimization of Water Networks*. Birkhäuser, Basel, 2012.

M. Meck, T. M. Müller, L. C. Altherr, and P. F. Pelz. Designing an efficient and cost-optimal industrial cooling system using mixed-integer nonlinear programming. In *Operations Research Proceedings 2019*, pages 505–512, Cham, 2020. Springer.

F. Meng, G. Fu, R. Farmani, C. Sweetapple, and D. Butler. Topological attributes of network resilience: A study in water distribution systems. *Water Res*, 143: 376–386, 2018.

A. Morsi, B. Geißler, and A. Martin. Mixed integer optimization of water supply networks. In *Mathematical Optimization of Water Networks*, pages 35–54. Birkhäuser, Basel, 2012.

T. M. Müller, L. C. Altherr, M. B. Ahola, S. Schabel, and P. F. Pelz. Multi-criteria optimization of pressure screen systems in paper recycling – balancing quality, yield, energy consumption and system complexity. In *EngOpt 2018 Proceedings of the 6th International Conference on Engineering Optimization*, pages 1216–1228, Cham, 2019a. Springer.

T. M. Müller, L. C. Altherr, M. B. Ahola, S. Schabel, and P. F. Pelz. Optimizing pressure screen systems in paper recycling: Optimal system layout, component selection and operation. In *Operations Research Proceedings 2018*, pages 355–361, Cham, 2019b. Springer.

T. M. Müller, P. Leise, T. Meck, L. C. Altherr, and P. F. Pelz. Systemic optimization of booster stations—from data collection to validation. In *4th International Rotating Equipment Conference—Pumps and Compressors*, Wiesbaden, 2019c.

T. M. Müller, P. Leise, I.-S. Lorenz, L. C. Altherr, and P. F. Pelz. Optimization and validation of pumping system design and operation for water supply in high-rise buildings. *Optim Eng*, 2020. doi: https://doi.org/10.1007/s11081-020-09553-4.

B. R. Munson, D. F. Young, T. H. Okiishi, and W. W. Huebsch. *Fundamentals of Fluid Mechanics.* John Wiley & Sons, Hoboken, 6th edition, 2009.

L. J. Murphy and A. R. Simpson. Genetic algorithms in pipe network optimisation. Technical Report Research Report R93, University of Adelaide, 1992.

G. L. Nemhauser and L. A. Wolsey. *Integer and Combinatorial Optimization.* Wiley, New York, 1988.

A. Ostfeld and A. Tubaltzev. Ant colony optimization for least-cost design and operation of pumping water distribution systems. *J Water Resour Plann Manag*, 134(2):107–118, 2008.

G. Pahl, W. Beitz, J. Feldhusen, and K.-H. Grote. *Engineering Design: A Systematic Approach.* Springer, London, 2007.

F. D. Paola, E. Galdiero, and M. Giugni. Location and setting of valves in water distribution networks using a harmony search approach. *J Water Resour Plann Manag*, 143(6):04017015, 2017.

N. R. Patel, J. B. Rawlings, M. J. Wenzel, and R. D. Turney. Design and application of distributed economic model predictive control for large-scale building temperature regulation. In *4th International High Performance Buildings Conference at Purdue 2016*, West Lafayette, 2016a. Ray W. Herrick Laboratories.

N. R. Patel, J. B. Rawlings, M. J. Wenzel, and R. D. Turney. Distributed economic model predictive control for large-scale building temperature regulation. In *2016 American Control Conference*, pages 895–900, Boston, 2016b.

F. Pecci, E. Abraham, and I. Stoianov. Global optimality bounds for the placement of control valves in water supply networks. *Optim Eng*, 20(2):457–495, 2019.

P. F. Pelz, U. Lorenz, T. Ederer, S. Lang, and G. Ludwig. Designing pump systems by discrete mathematical topology optimization: The artificial fluid systems designer (AFSD). In *International Rotating Equipment Conference (IREC) 2012*, Düsseldorf, 2012.

M. E. Pfetsch, A. Fügenschuh, B. Geißler, N. Geißler, R. Gollmer, B. Hiller, J. Humpola, T. Koch, T. Lehmann, A. Martin, A. Morsi, J. Rövekamp, L. Schewe, M. Schmidt, R. Schultz, R. Schwarz, J. Schweiger, C. Stangl, M. C. Steinbach, S. Vigerske, and B. M. Willert. Validation of nominations in gas network optimization: Models, methods, and solutions. *Optimization Methods and Software*, 30(1):15–53, 2015.

P. Pöttgen and P. F. Pelz. Evaluation of different approaches for the optimization of layout and control of booster stations. In *ECCOMAS Congress 2016—VII European Congress on Computational Methods in Applied Sciences and Engineering*, Crete Island, 2016.

P. Pöttgen and P. F. Pelz. The best attainable EEI for booster stations derived by global optimization. In *International Rotating Equipment Conference (IREC) 2016*, Düsseldorf, 2016.

P. Pöttgen, T. Ederer, L. C. Altherr, and P. F. Pelz. Developing a control strategy for booster stations under uncertain load. In *Uncertainty in Mechanical Engineering II*, pages 241–246, Switzerland, 2015. Trans Tech Publications Ltd.

P. Pöttgen, T. Ederer, L. C. Altherr, U. Lorenz, and P. F. Pelz. Examination and optimization of a heating circuit for energy-efficient buildings. *Energy Technol*, 4(1):136–144, 2016.

K. M. Powell, W. J. Cole, U. F. Ekarika, and T. F. Edgar. Optimal chiller loading in a district cooling system with thermal energy storage. *Energy*, 50:445–453, 2013.

R. Rahmaniani, T. G. Crainic, M. Gendreau, and W. Rei. The benders decomposition algorithm: A literature review. *Eur J Oper Res*, 259(3):801–817, 2017.

L. Rausch. *Mathematische Optimierung von Wasserversorgungsnetzen für informelle Siedlungen in Mega Cities [Mathematical optimization of water supply networks for informal settlements in mega cities]*. Dissertation, Technische Universität Darmstadt, 2018.

L. Rausch, P. Leise, T. Ederer, L. C. Altherr, and P. F. Pelz. A comparison of MILP and MINLP solver performance on the example of a drinking water supply system design problem. In *ECCOMAS Congress 2016—VII European Congress on Computational Methods in Applied Sciences and Engineering*, Crete Island, 2016.

J. B. Rawlings, N. R. Patel, M. J. Risbeck, C. T. Maravelias, M. J. Wenzel, and R. D. Turney. Economic MPC and real-time decision making with application to large-scale HVAC energy systems. *Comput Chem Eng*, 114:89–98, 2018.

C. Reintjes and U. Lorenz. Bridging mixed integer linear programming for truss topology optimization and additive manufacturing. *Optim Eng*, 2020. doi: https://doi.org/10.1007/s11081-020-09541-8.

L. F. R. Reis, R. M. Porto, and F. H. Chaudhry. Optimal location of control valves in pipe networks by genetic algorithm. *J Water Resour Plann Manag*, 123(6): 317–326, 1997.

J. Riordan and C. E. Shannon. The number of two-terminal series-parallel networks. *J Math Phys*, 21:83–92, 1942.

R. Ríos-Mercado and C. Borraz-Sánchez. Optimization problems in natural gas transportation systems: A state-of-the-art review. *Applied Energy*, 147:536–555, 2015.

M. J. Risbeck. *Mixed-Integer Model Predictive Control with Applications to Building Energy Systems.* Dissertation, University of Wisconsin–Madison, 2018.

M. J. Risbeck, C. T. Maravelias, J. B. Rawlings, and R. D. Turney. Cost optimization of combined building heating/cooling equipment via mixed-integer linear programming. In *2015 American Control Conference*, pages 1689–1694, Chicago, 2015.

M. J. Risbeck, C. T. Maravelias, J. B. Rawlings, and R. D. Turney. A mixed-integer linear programming model for real-time cost optimization of building heating, ventilation, and air conditioning equipment. *Energy Build*, 142:220–235, 2017.

P. Sanders. Algorithm engineering—an attempt at a definition. In *Efficient Algorithms*, pages 321–340. Springer, Berlin, Heidelberg, 2009.

B. Saul, P. Pöttgen, W. Zimmermann, and P. F. Pelz. SHEP: An innovative language to create and evaluate optimization programs for pump systems. In *International Rotating Equipment Conference (IREC) 2016*, Düsseldorf, 2016.

D. Savić and G. A. Walters. Genetic algorithms for least-cost design of water distribution networks. *J Water Resour Plan Manag*, 123(2):67–77, 1997.

C. Schänzle, L. C. Altherr, T. Ederer, U. Lorenz, and P. F. Pelz. As good as it can be—ventilation system design by a combined scaling and discrete optimization method. In *FAN 2015—Proceedings of the International Conference on Fan Noise, Aerodynamics, Applications and Systems*, Lyon, 2015.

A. Schrijver. *Theory of Linear and Integer Programming.* Wiley, New York, 1986.

T. Schütz, R. Streblow, and D. Müller. A comparison of thermal energy storage models for building energy system optimization. *Energ Buildings*, 93:23–31, 2015.

T. Schütz, H. Harb, M. Fuchs, and D. Mueller. Optimal design of building energy systems for residential buildings. In *CLIMA 2016 - 12th REHVA World Congress*, Aalborg, 2016. Department of Civil Engineering, Aalborg University.

H. Sun, L. C. Altherr, J. Pei, P. F. Pelz, and S. Q. Yuan. Optimal booster station design and operation under uncertain load. In *Uncertainty in Mechanical Engineering III*, pages 102–115, Switzerland, 2018. Trans Tech Publications Ltd.

Y.-H. Sung, M.-D. Lin, Y.-H. Lin, and Y.-L. Liu. Tabu search solution of water distribution network optimization. *J Environ Eng Manag*, 17:177–187, 2017.

S. T. Taylor. Primary-only vs. primary-secondary variable flow systems. *ASHRAE J*, 44(2):25–29, 2002.

S.-T. Tillmann and C.-A. Bohn. Simulation of water condensation based on a thermodynamic approach. In *VMV '15: Proceedings of the Conference on Vision, Modeling and Visualization*, Aachen, 2015. Eurographics Association.

D. J. Tritton. *Physical Fluid Dynamics.* Oxford Science Publications, Oxford, 1988.

B. Ulanicki, J. Kahler, and B. Coulbeck. Modeling the efficiency and power characteristics of a pump group. *Modeling the Efficiency and Power Characteristics of a Pump Group*, 134(1):88–93, 2008.

A.-J. Ulusoy, F. Pecci, and I. Stoianov. An MINLP-based approach for the design-for-control of resilient water supply systems. *IEEE Syst J*, 14(3):4579–4590, 2020.

United Nations Framework Convention on Climate Change (UNFCCC). Report of the conference of the parties on its twenty-first session, held in Paris from 30 November to 13 December 2015. Addendum. Part two: Action taken by the conference of the parties at its twenty-first session, 2016. https://unfccc.int/sites/default/files/resource/docs/2015/cop21/eng/10a02.pdf (accessed May 07, 2021).

VDI 2221. Design of technical products and systems; VDI guidelines, Nov. 2019.

VDI 2222. Methodic development of solution principles; VDI guidelines, June 1997.

VDI 2223. Systematic embodiment design of technical products; VDI guidelines, Jan. 2004.

VDI 2884. Purchase, operating and maintenance of production equipment using life cycle costing (LCC); VDI guidelines, Dec. 2015.

D. Verleye and E.-H. Aghezzaf. Optimising production and distribution operations in large water supply networks: A piecewise linear optimisation approach. *Int J Prod Res*, 51(23-24):7170–7189, 2013.

J. P. Vielma. Mixed integer linear programming formulation techniques. *SIAM Rev*, 57(1):3–57, 2015.

J. P. Vielma and G. Nemhauser. Modeling disjunctive constraints with a logarithmic number of binary variables and constraints. *Math Program*, 128:199–213, 2011.

J. P. Vielma, S. Ahmed, and G. Nemhauser. Mixed-integer models for nonseparable piecewise-linear optimization: Unifying framework and extensions. *Oper Res*, 58 (2):303–315, 2010.

P. Voll. *Automated Optimization-Based Synthesis of Distributed Energy Supply Systems*. Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, 2013.

P. Voll, M. Lampe, G. Wrobel, and A. Bardow. Superstructure-free synthesis and optimization of distributed industrial energy supply systems. *Energy*, 45(1):424–435, 2012.

P. Voll, C. Klaffke, M. Hennen, and A. Bardow. Automated superstructure-based synthesis and optimization of distributed energy supply systems. *Energy*, 50: 374–388, 2013.

M. Walmsley, M. Atkins, and J. Riley. Thermocline management of stratified tanks for heat storage. *Chem Eng Trans*, 18:231–236, 2009.

J. B. Weber and U. Lorenz. Optimizing booster stations. In *GECCO '17: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1303–1310, New York, 2017. ACM.

J. B. Weber and U. Lorenz. Algorithmic system design of thermofluid systems. In *EngOpt 2018 Proceedings of the 6th International Conference on Engineering Optimization*, pages 132–143, Cham, 2019a. Springer.

J. B. Weber and U. Lorenz. Modeling thermofluid systems: An approach customized for optimization. In *Operations Research Proceedings 2018*, pages 387–393, Cham, 2019b. Springer.

J. B. Weber, M. Hartisch, A. Herbst, and U. Lorenz. Towards an algorithmic synthesis of thermofluid systems. *Optim Eng*, 2020a. doi: https://doi.org/10.1007/s11081-020-09564-1.

J. B. Weber, M. Hartisch, and U. Lorenz. Optimized design of thermofluid systems using the example of mold cooling in injection molding. In *Operations Research Proceedings 2019*, pages 473–480, Cham, 2020b. Springer.

R. J.-B. Wets. Stochastic programs with fixed recourse: The equivalent deterministic program. *SIAM Rev*, 16(3):309–339, 1974.

F. M. White. *Fluid Mechanics*. McGraw-Hill series in mechanical engineering. McGraw Hill, Boston, 2011.

Wilo SE. Wilo-Economy MHIE, 2020a. https://wilo.com/lb/en/Products-and-expertise/Series-Finder/Wilo-Economy-MHIE_121.html (accessed May 07, 2021).

Wilo SE. Economy MHIE 203N, 2020b. https://wilo.com/lb/en/Products-and-expertise/Series-Finder/Economy-MHIE-203N-(3~380-400-440-V-EPDM)_6598.html (accessed May 07, 2021).

J. Wolf. *Quantified Linear Programming*. Dissertation, Technische Universität Darmstadt, 2015.

L. A. Wolsey. *Integer Programming*. Wiley Series in Discrete Mathematics and Optimization. Wiley, New York, 1998.

R. Yokoyama, Y. Hasegawa, and K. Ito. A MILP decomposition approach to large scale optimization in structural design of energy supply systems. *Energy Convers Manag*, 43(6):771–790, 2002.

A. C. Zecchin, H. R. Maier, A. R. Simpson, M. Leonard, and J. B. Nixon. Ant colony optimization applied to water distribution system design: Comparative study of five algorithms. *J Water Resour Plan Manag*, 133:87–92, 2007.

D. Zhang, P. B. Luh, J. Fan, and S. Gupta. Chiller plant operation optimization: Energy-efficient primary-only and primary-secondary systems. *IEEE Trans Autom Sci Eng*, 15(1):341–355, 2017.