

Entwicklung von Konzepten zur optimalen Regelung verteilter regenerativer Wärmeeinspeisung in Fernwärmenetzen

Oliver Feindt



Universität
Bremen

2023

Entwicklung von Konzepten zur optimalen Regelung verteilter regenerativer Wärmeeinspeisung in Fernwärmenetzen

Vom Fachbereich für Physik und Elektrotechnik
der Universität Bremen

zur Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

genehmigte Dissertation

von

M. Sc. Oliver Feindt

wohnhaft in Bremen

Referent:	Prof. Dr.-Ing. Kai Michels
Korreferent:	Prof. Dr.-Ing. Andreas Nürnberger
Eingereicht am:	21.02.2023
Tag des Promotionskolloquiums:	23.06.2023

Zusammenfassung

In dieser Arbeit wird ein Automatisierungskonzept für den optimalen Betrieb von Fernwärmenetzen untersucht. Ziel ist sowohl der wirtschaftlich optimale Betrieb als auch die Einhaltung aller thermo-hydraulischen Restriktionen der im Rahmen der Energiewende komplexer werdenden Fernwärmeversorgung.

Zunächst wird der theoretische Hintergrund von Fernwärmenetzen und deren Modellierung sowie Optimierung und Regelung dargelegt. Basierend auf den Erkenntnissen wird ein Automatisierungskonzept vorgeschlagen, welches den Betrieb bestehender Fernwärmenetze mit wirtschaftlicher Optimierung um eine Ebene mit Netzsimulation und Modellprädiktiver Regelung erweitert.

Die Entwicklung einer geeigneten Netzsimulation ist Gegenstand der Untersuchungen. Physikalische Modelle von vier Fernwärmenetzen werden erstellt und durch Ansätze aus den Bereichen des maschinellen Lernens und der Systemidentifikation vereinfacht. Dies führt zu deutlich reduzierten Rechenzeiten und einer guten Simulationsgenauigkeit verglichen mit den vollständigen physikalischen Modellierungen.

Anschließend wird die Optimierungskomponente einer Modellprädiktiven Regelung mit den vereinfachten Modellen entwickelt. Leistungen und Vorlauftemperaturen aller Wärmeerzeugungsanlagen sollen möglichst nah am wirtschaftlichen Optimum eingestellt werden, während alle thermo-hydraulischen Restriktionen wie minimale oder maximale Drücke und Temperaturen in den Netzsimulationen eingehalten werden. Die Untersuchungen zeigen, dass einfache Netze in Sekunden und komplexere Netze in etwa einer Minute optimiert werden können. Dabei werden hauptsächlich stationäre Simulationen verwendet, welche gewisse dynamische Effekte im Netz nicht berücksichtigen können. Die Anwendbarkeit der stationär optimierten Fahrweisen wird deshalb durch instationäre Simulationen überprüft. Ebenfalls wird grundlegend eine Optimierung unter Verwendung von instationären Simulationen getestet. Abschließend werden die Limitierungen des vorgestellten Konzepts sowie der erreichte Entwicklungsstand diskutiert.

Abstract

This thesis investigates an automation concept for the optimal operation of district heating networks. The aim is to achieve both an economically optimal operation and compliance with all thermo-hydraulic constraints of district heating systems, which are becoming increasingly complex in the context of the energy transition.

First, the theoretical background of district heating networks and their modelling, optimisation and control is presented. Based on the findings, an automation concept is proposed that adds a level of network simulation and Model Predictive Control to the operation of existing district heating networks with economic optimisation.

The next step is to develop a suitable network simulation. Physical models of four district heating networks are created and simplified using machine learning and system identification approaches. This leads to significantly reduced computation times and good simulation accuracy compared to full physical modelling.

Finally, the optimisation component of a Model Predictive Control is developed using the simplified models. The aim is to set plant power outputs and supply temperatures as close to the economic optimum as possible, while respecting all thermo-hydraulic constraints such as minimum or maximum pressures and temperatures. The studies show that simple networks can be optimised in seconds and more complex networks in about a minute. Steady-state simulations are mainly used, which cannot account for some dynamic effects in the network. Therefore, the applicability of the steady-state optimised operation is verified by dynamic simulations. Furthermore, a basic optimisation using dynamic simulations is tested. Finally, the limitations of the presented concept as well as the achieved development status are discussed.

Vorwort

An dieser Stelle möchte ich mich bei allen bedanken, die mich bei der Erstellung dieser Dissertation unterstützt haben. Diese entstand während meiner Tätigkeit als Wissenschaftlicher Mitarbeiter am Institut für Automatisierungstechnik des Fachbereichs für Physik und Elektrotechnik an der Universität Bremen.

Mein besonderer Dank gilt Herrn Prof. Dr.-Ing. Kai Michels¹, der mir diese Arbeit nicht nur ermöglicht, sondern mich stets bei allen Fragen und Problemen enorm unterstützt sowie mir alle wissenschaftlichen Freiheiten gelassen hat. Zudem danke ich Herrn Dr.-Ing. Pierl¹, Herrn Dr.-Ing. Schüttler¹ und Herrn Simon Fleischer¹, die mir ebenfalls sehr geholfen haben und immer für eine Diskussion offen waren sowie allen meinen Kolleginnen und Kollegen am Institut für Automatisierungstechnik für den fachlichen und persönlichen Austausch.

Mein Dank gilt weiterhin Herrn Prof. Dr.-Ing. Andreas Nürnberger² und seiner Data and Knowledge Engineering Group der Otto-von-Guericke-Universität Magdeburg für die konstruktive Kritik und Unterstützung bei der Erstellung und Validierung von neuronalen Netzen sowie für die Übernahme eines Gutachtens.

Auch den Stadtwerken München als Industriepartner und im Besonderen Frau Dr.-Ing. Karina Nold, Herrn Michael Sachsenhauser und Herrn Dominik Hülsmeier gilt mein Dank für die stets sehr gute Zusammenarbeit, die offenen Diskussionen über organisatorische und fachliche Themen sowie die Bereitstellung von Daten und Software. An dieser Stelle möchte ich auch den Herren Georg Käser, Dr. Michael Fischer, Dr.-Ing. Andreas Wolters und Wolfgang Micus für die komplexe Modellierung der Fernwärmenetze, die tatkräftige Unterstützung bei Fragen der Modellbildung und die Bereitstellung von Forschungslizenzen der Software *SIR 3S*[®] danken. Mein Dank gilt zudem Herrn Friedrich Fischer-Uhrig für die Nutzung der Software *STANET*[®] *Netzberechnung* als Referenz für die Modellierung in dieser Arbeit.

Weiterhin spreche ich dem *Projekträger Jülich* und dem *Bundesministerium für Wirtschaft und Klimaschutz* meinen Dank für die finanzielle Förderung des Forschungsvorhabens aus, welche die hier vorliegende Arbeit möglich gemacht hat.

Nicht zuletzt möchte ich vor allem meiner Familie, meiner Partnerin sowie allen meinen Freunden danken, die mich während meiner Promotion begleitet und mich nicht nur persönlich, sondern auch fachlich unterstützt haben.

Oliver Feindt

¹Institut für Automatisierungstechnik, Universität Bremen

²Data and Knowledge Engineering Group, Otto-von-Guericke-Universität Magdeburg

Liste der Veröffentlichungen

Die folgenden Arbeiten wurden im Rahmen dieser Dissertation veröffentlicht:

NOLD, K. ; FEINDT, O. : *Vorstellung des F&E Verbundprojekts GEOmaRe - Teilprojekt GEOMATIC*. Der Geothermiekongress, München, November 2019

PFRANG, D. ; DIRNER, S. ; DORSCH, K. ; BOHNSACK, D. ; BEICHEL, K. ; FEINDT, O. ; HEINE, F. ; HÖRBRAND, T. ; LENTSCH, D. ; MEINECKE, M. ; NOLD, K. ; PLOTTEN, M. ; SAVVATIS, A. ; SCHÖLDERLE, F. ; WINTER, T. ; ZOSSEDER, K. : *GEOmaRe - Ganzheitlich optimierte und nachhaltige Reservoirerschließung für tiefengeothermische Anlagen im bayerischen Molassebecken*. Geothermische Energie, Heft 97, 2020

FEINDT, O. ; KÄSER, G. ; FISCHER, M. ; DIRNER, S. ; NOLD, K. ; SACHSENHAUSER, M. ; MICHELS, K. : *GEOMATIC - Optimale Verteilung der Wärmeeinspeisung in Fernwärmenetzen*. Euro Heat & Power, Heft 4-5, 2022. - ISSN 0949-166X-D9790F

FEINDT, O. ; SACHSENHAUSER, M. ; NOLD, K. ; DIRNER, S. ; KÄSER, G. ; FISCHER, M. ; MICHELS, K. : *Modellprädiktive Regelung für die optimale Verteilung der Wärmeeinspeisung in Fernwärmenetzen*. 54. Kraftwerkstechnisches Kolloquium, Dresden. In: M. Beckmann, A. Hurtado (Hrsg.), Kraftwerkstechnik 2022 - Power Plant Technology. Innovation- und Kreislaufwirtschaft Sachsen e. V., Freiberg, 2022. - ISBN 978-3-00-072104-5

Inhaltsverzeichnis

1. Einleitung	1
1.1. Zielsetzung	2
1.2. Struktur und Methodik	2
2. Grundlagen	5
2.1. Aufbau und Funktionsweise von Fernwärmenetzen	5
2.1.1. Wärmeerzeugungsanlagen	6
2.1.2. Pumpen	7
2.1.3. Ventile und Absperrarmaturen	8
2.1.4. Rohrleitungen	8
2.1.5. Wärmetauscher	10
2.2. Grundlagen der Optimierung	11
2.2.1. Zielfunktion und Nebenbedingungen	11
2.2.2. Klassifikation von Optimierungsproblemen	11
2.2.3. Optimalitätsbedingungen	12
2.2.4. Lokale Optimierung ohne Nebenbedingungen	14
2.2.5. Lokale Optimierung mit Nebenbedingungen	16
2.2.6. Globale Optimierung	18
2.3. Modellprädiktive Regelung	19
2.4. Systemidentifikation	21
2.4.1. Auswahl der Modellstruktur	21
2.4.2. Berechnung der Modellparameter	25
2.5. Maschinelles Lernen	26
2.5.1. Algorithmen des maschinellen Lernens	28
2.5.2. Lineare Modelle	28
2.5.3. Entscheidungsbäume	29
2.5.4. Neuronale Netze	29
2.5.5. Ensemble Learning	33
2.5.6. Bestärkendes Lernen	33
3. Stand der Technik	35
3.1. Modellierung von Fernwärmenetzen	35
3.1.1. Physikalische Netzmodelle	36
3.1.2. Modellierung durch Ansätze des maschinellen Lernens	38
3.2. Betrieb von Fernwärmenetzen	38
3.3. Optimierung und Regelung von Fernwärmenetzen	41
3.3.1. Optimale Fahrweise von Fernwärmenetzen	41
3.3.2. Überblick relevanter Forschungsprojekte	44
4. Regelungskonzept	51
4.1. Derzeitiger Betrieb der untersuchten Fernwärmenetze	51
4.2. Konzept für den optimalen Betrieb von Fernwärmenetzen	52
4.2.1. Anforderungen an die Optimierung und Netzsimulation	54

4.2.2.	Vorgehensweise	55
4.2.3.	Limitierungen des Konzepts	56
5.	Thermo-hydraulische Modellierung von Fernwärmenetzen	57
5.1.	Stationäre hydraulische Berechnung	57
5.2.	Stationäre thermische Berechnung	63
5.3.	Instationäre thermische Berechnung	67
5.4.	Modellierung in <i>Simulink</i>	69
5.5.	Knoten-Kanten-Modelle	71
5.5.1.	Knoten-Kanten-Matrix	71
5.5.2.	Hydraulische Netzberechnung	72
5.5.3.	Thermische Netzberechnung	77
5.5.4.	Zusammenführung der hydraulischen und thermischen Netzberechnung	79
5.5.5.	Kommerzielle Programme	80
6.	Modellbildung- und auswertung für vier Beispielnetze	81
6.1.	Fernwärmenetz 1	82
6.1.1.	Vorauswahl der Modellierungsansätze	82
6.1.2.	Netz 1: <i>SIR 3S</i> [®]	84
6.1.3.	Netz 1: Vereinfachte stationäre Modelle	86
6.1.4.	Netz 1: Vereinfachte instationäre Modelle	98
6.2.	Fernwärmenetz 2	103
6.2.1.	Netz 2: <i>SIR 3S</i> [®]	103
6.2.2.	Netz 2: Maschinelles Lernen	104
6.2.3.	Netz 2: Vorhersage der Rücklauftemperatur	113
6.3.	Fernwärmenetz 3	115
6.3.1.	Netz 3: <i>SIR 3S</i> [®]	115
6.3.2.	Netz 3: Neuronale Netze	116
6.4.	Fernwärmenetz 4	124
6.5.	Diskussion der Modellierungsergebnisse	126
7.	Optimierung der Fernwärmenetze	129
7.1.	Optimierung von Fernwärmenetz 1	130
7.1.1.	Netz 1: Optimierung der Einspeiseleistungen	131
7.1.2.	Netz 1: Stationäre Optimierung	137
7.1.3.	Netz 1: Instationäre Optimierung	140
7.2.	Optimierung von Fernwärmenetz 2	143
7.2.1.	Optimierung von Netz 2 nach Leistungsprognose und wirtschaftlichen Sollwerten	144
7.2.2.	Optimierung von Netz 2 mit Einspeiseleistung aus Prozessdaten und wirtschaftlichen Sollwerten	150
7.2.3.	Netz 2: Verwendung der Vorlaufleitungen als Temperaturspeicher	154
7.2.4.	Minimierung der Wärmeverluste in Fernwärmenetz 2	154
7.3.	Optimierung von Fernwärmenetz 3	156
7.3.1.	Fahrweise der Druckerhöhungsanlage	157
7.3.2.	Optimierung von Netz 3 nach Leistungsprognose und wirtschaftlichen Sollwerten	158
7.3.3.	Optimierung von Netz 3 mit Einspeiseleistung aus Prozessdaten und wirtschaftlichen Sollwerten	164
7.3.4.	Netz 3: Verwendung der Vorlaufleitungen als Temperaturspeicher	166
7.3.5.	Minimierung der Wärmeverluste in Fernwärmenetz 3	166
7.4.	Optimierung von Fernwärmenetz 4	166
7.5.	Diskussion der Optimierungsergebnisse	167

8. Fazit und Ausblick	169
A. Anhang	171
A.1. Weitere Informationen zur Modellbildung und -auswertung	171
A.1.1. Programmcode: Neuronale Netze	171
A.1.2. Modellvalidierung Netz 1	175
A.1.3. Modellvalidierung Netz 2	181
A.1.4. Modellvalidierung Netz 3	190
A.1.5. Modellbildung und -auswertung - Fernwärmenetz 4	202
A.2. Optimierung von Fernwärmenetz 4	212
B. Symbolverzeichnis	219
C. Abkürzungsverzeichnis	221
D. Abbildungsverzeichnis	223
E. Tabellenverzeichnis	227
F. Literaturverzeichnis	231

1. Einleitung

Im Rahmen der Energiewende wird die Wärmeerzeugung in Fernwärmenetzen auf regenerative Energien umgestellt. Diese Umstellung erfordert eine zunehmend dezentrale Wärmeeinspeisung mehrerer kleiner Wärmeerzeugungsanlagen und erhöht somit die Komplexität der Betriebsweise und Regelung der Netze. In dieser Arbeit werden Konzepte entwickelt, um diese Problematik zu lösen und die dezentrale Wärmeeinspeisung in Fernwärmenetzen optimal zu regeln.

Fernwärme deckt in Deutschland etwa 8% [11] des Wärmebedarfs und wird bisher hauptsächlich in Heizkraftwerken (HKW), welche durch Kraft-Wärme-Kopplung (KWK) aus Kohle oder Gas Elektrizität und Wärme erzeugen, gewonnen. Bis spätestens 2038 soll die Energieversorgung durch Kohlekraftwerke in Deutschland nach dem Kohleausstiegsgesetz (BGBI. I S. 1818 [1]) beendet werden. Im Rahmen dieses Gesetzes werden neben dem Kohleausstieg KWK-Anlagen und die Wärmeerzeugung durch erneuerbare Energien weiter gefördert. Für Fernwärmenetze wird also auch eine regenerative Wärmeerzeugung durch Energiequellen wie zum Beispiel Geothermie, Biomasse oder Solarenergie geschaffen. Bisher ist der Anteil erneuerbarer Energien mit 15,6% an der Wärme- und Kälteerzeugung im Jahr 2020 gering [11, S. 15].

Die Regelung von großen Fernwärmenetzen ist schon heute komplex. Durch gewachsene Strukturen der Netze gibt es viele Restriktionen, die eingehalten werden müssen, um jedem Wärmeabnehmer die Wärme mit ausreichend hohen Temperaturen und Drücken zur Verfügung zu stellen. Dabei erfolgt die Wärmeversorgung heute noch weitestgehend zentral. Ein einzelner Kraftwerksstandort liefert den Großteil der Wärme und kann das Netz und auftretende Schwankungen stabil ausregeln. Zu Zeiten großer Wärmeabnahmen werden kleinere Heizwerke hinzugeschaltet. Die Auswahl, welche Kraftwerke Wärme in das System einspeisen, erfolgt manuell und teilweise mit Sollwerten aus wirtschaftlichen Optimierungen.

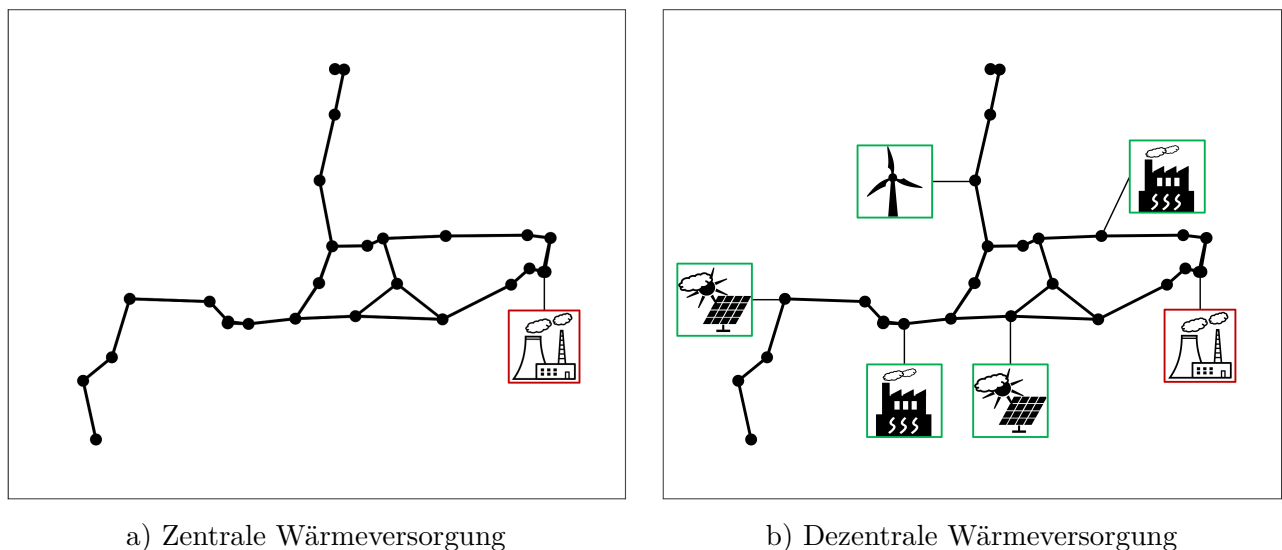


Abbildung 1.1.: Beispiel eines Fernwärmenetzes mit zentraler und dezentraler Wärmeversorgung.

Bei einer regenerativen Wärmeversorgung großer Fernwärmenetze wird davon ausgegangen, dass mehrere kleine Wärmeerzeugungs- sowie Speicheranlagen und eine dezentrale Wärmeeinspeisung erforder-

derlich werden. Abbildung 1.1 zeigt beispielhaft ein Fernwärmenetz mit zentraler und ein Netz mit dezentraler Wärmeversorgung. Durch die Dezentralisierung wird die Regelung komplexer und es wird in den gewachsenen Strukturen der Netze schwieriger, beispielsweise sämtliche hydraulische Restriktionen einzuhalten. Die Netze können nicht mehr manuell gefahren werden. Es ist somit von Interesse, ein System zu entwickeln, welches die Kraftwerke automatisch steuert und die Temperaturen und Drücke im Netz regelt.

Die vorliegende Arbeit beschäftigt sich mit der Entwicklung eines solchen Systems am Beispiel vier zunehmend komplexer Fernwärmenetze. Die Arbeit wurde im Rahmen des Verbundvorhabens GEOmaRe (Verbundvorhaben: GEOmaRe - Optimierte Regel- und Anlagentechnik mit nachhaltiger Reservoirbewirtschaftung für die tiefengeothermischen Wärmeprojekte im Münchener Raum; Teilvorhaben: ReGEO ' Reservoirengineering und GEOMATIC, Förderkennzeichen 0324332A) des Forschungszentrums Jülich GmbH zusammen mit den Stadtwerken München (SWM Services GmbH) durchgeführt. Ziel des Verbundvorhabens ist die konzeptionelle, entwicklungstechnische und wissenschaftliche Begleitung des Geothermieausbaus auf 400 MW in der Stadt München. In diesem Rahmen wurden im obertägigen Teilvorhaben GEOMATIC unter anderem Steuerungs- und Regelungstechnik sowie Wartungskonzepte und Betriebsmodelle erarbeitet. Ziele des Teilvorhabens GEOMATIC wurden in [91] und [99] vorgestellt. Zudem wurden Teilergebnisse dieser Dissertation in [39] und [40] veröffentlicht.

1.1. Zielsetzung

Im Vorhaben GEOMATIC wird die Einbindung zusätzlicher Geothermieanlagen in das Regelungskonzept von Fernwärmenetzen untersucht. In diesem Kontext stellt sich allgemein die Frage der optimalen regelungstechnischen Einbindung dezentraler Energiequellen in ein Fernwärmenetz. Ziel dieser Promotion ist es, ein Regelungskonzept für den optimalen Betrieb von Fernwärmenetzen mit dezentraler regenerativer Wärmeeinspeisung zu entwickeln.

Grundlage für die Entwicklung einer Regelung ist ein Simulationsmodell, welches Druck-, Durchfluss- und Temperaturwerte sowie die notwendigen Restriktionen für eine Optimierung berechnen und prognostizieren kann. Mithilfe dieses Modells soll ein Algorithmus entwickelt werden, welcher je nach Lastzustand die Wärmeerzeugungsanlagen und deren Einspeiseleistungen sowie Vorlauftemperaturen auswählt. Die Auswahl der Anlagen, Leistungen und Vorlauftemperaturen soll sowohl wirtschaftliche Kriterien als auch die hydraulischen und thermischen Restriktionen der Fernwärmenetze berücksichtigen. Die Kriterien und Restriktionen des optimalen Betriebs sollen in dieser Arbeit herausgestellt werden.

Die Optimierung von Fernwärmenetzen kann dabei nicht separat betrachtet werden, da Fernwärmenetze häufig beispielsweise durch KWK-Anlagen mit anderen Systemen wie Strom- oder Fernkältenetzen gekoppelt sind. Auch zusätzliche Verbindungen mehrerer Fernwärmenetze sind möglich. Das Optimierungsproblem kann daher auf verschiedenen Komplexitätsebenen betrachtet werden. Der genaue Umfang der zu entwickelnden Optimierung wird in dieser Arbeit bestimmt. Eine Optimierung des Gesamtsystems ist für die Automatisierung erforderlich, kann aber auf verschiedene Systeme verteilt werden.

1.2. Struktur und Methodik

Die Arbeit beginnt in Kapitel 2 mit der Beschreibung technischer und theoretischer Grundlagen. Zunächst wird dabei in Kapitel 2.1 auf den Aufbau von Fernwärmenetzen eingegangen. Das in dieser

Arbeit entwickelte Regelungskonzept basiert auf Optimierungen und dem Ansatz der Modellprädiktiven Regelung. Deren Theorie wird in den Kapiteln 2.2 und 2.3 beschrieben. Zudem werden Methoden der Systemidentifikation und des maschinellen Lernens in den Kapiteln 2.4 und 2.5 vorgestellt, welche später für die Erstellung vereinfachter mathematischer Modelle von Fernwärmenetzen verwendet werden.

Anschließend folgt in Kapitel 3 der Stand der Technik. Kapitel 3.1 geht dabei zunächst auf thermohydraulische Fernwärmenetzmodelle ein und Kapitel 3.2 beschreibt den Betrieb von Fernwärmenetzen. Im letzten Teil 3.3 werden die optimale Fahrweise von Fernwärmenetzen beschrieben sowie relevante Projekte der Optimierung und Regelung zusammengefasst.

Basierend auf den Grundlagen und dem Stand der Technik wird in Kapitel 4 ein Optimierungs- und Regelungskonzept entwickelt. Dabei wird zunächst auf den aktuellen Aufbau und Betrieb der hier analysierten Fernwärmenetze eingegangen und anschließend ein Konzept vorgeschlagen. Anforderungen an das Konzept und die Modellierung sowie die weitere Vorgehensweise und Einschränkungen dieser Arbeit werden erläutert.

Die Modellierung von Fernwärmenetzen wird in Kapitel 5 beschrieben. Physikalische Modelle für hydraulisch stationäre sowie thermisch stationäre und instationäre Simulationen werden in Kapitel 5.1 bis 5.5 entwickelt und kommerzielle Umsetzungen vorgestellt. Zudem werden vereinfachte Modelle unter Verwendung von Methoden der Systemidentifikation und des maschinellen Lernens erstellt. Die Modelle von vier zunehmend komplexen Fernwärmenetzen werden in Kapitel 6 hinsichtlich ihrer Anwendung in dem zu entwickelnden Konzept verglichen. Modelle werden ausgewertet und für den Einsatz in einer Optimierung ausgewählt.

In Kapitel 7 wird das Optimierungskonzept auf die vier Fernwärmenetze angewandt. Ergebnisse, Limitierungen und offene Punkte der Optimierung werden präsentiert. Zuletzt werden in Kapitel 8 ein Fazit sowie ein Ausblick auf weiterführende Forschungsthemen gegeben.

2. Grundlagen

In diesem Kapitel werden technische und theoretische Grundlagen vorgestellt, welche im weiteren Verlauf der Arbeit benötigt werden. Zunächst wird in Kapitel 2.1 der Aufbau und die Funktionsweise von Fernwärmenetzen beschrieben. Die Kapitel 2.2 und 2.3 befassen sich mit Grundlagen der Optimierung und Modellprädiktiven Regelung. Anschließend geht es in Kapitel 2.4 und 2.5 um die Grundlagen der Systemidentifikation und des maschinellen Lernens. Die beiden Methoden sind in der Lage, basierend auf Mess- oder Simulationsdaten vereinfachte mathematische Berechnungsmodelle zu erstellen.

2.1. Aufbau und Funktionsweise von Fernwärmenetzen

Fernwärmenetze sind thermo-hydraulische Systeme, welche großflächig Haushalte und Industrie mit Wärme versorgen. Abbildung 2.1 zeigt grundlegend den Aufbau eines Fernwärmenetzes. Die Wärme wird von mindestens einer Wärmequelle, der Wärmeinspeisung, konvektiv durch Rohre zu den Verbrauchern und anschließend wieder zurücktransportiert.

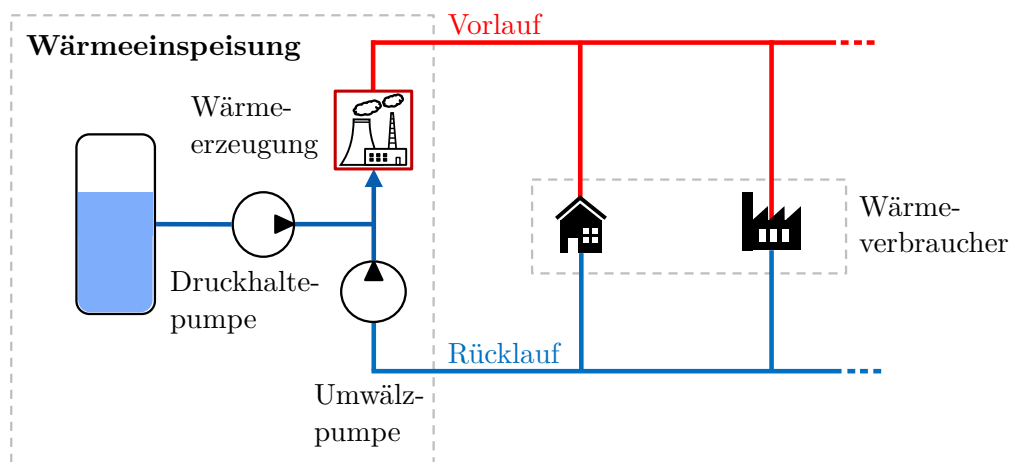


Abbildung 2.1.: Schematischer Aufbau eines Fernwärmenetzes nach [100].

Das Transportmedium in modernen Fernwärmenetzen ist Wasser bei Temperaturen von bis zu 140 °C. Ältere Netze werden teilweise mit noch höheren Temperaturen von 180 °C betrieben [61, S. 9]. Das Druckniveau in Fernwärmenetzen ist entsprechend hoch, um Verdampfen des Wassers und Kavitation in den Pumpen zu verhindern. In älteren Fernwärmenetzen oder für die Versorgung mit Prozesswärme in Industrienetzen wird noch Wasserdampf als Transportmedium verwendet. Europäische Fernwärmenetze werden aber nach und nach auf Wasser als Transportmedium umgestellt [92, S. 66]. In der vorliegenden Arbeit werden deshalb nur Heißwassernetze betrachtet.

Der Transport des heißen Wassers zum Verbraucher erfolgt im Vorlauf. Die Verbraucher entnehmen dem Wasser über Wärmetauscher Energie und kühlen es dadurch ab. Anschließend wird das kalte Wasser im Rücklauf zur Wärmequelle zurücktransportiert und dort wieder erwärmt. Für die Förderung des Wassers in den Rohrleitungen ist ein Druckunterschied zwischen Vor- und Rücklauf notwendig,

welcher durch Umwälzpumpen hergestellt wird. Der höchste Druck im Netz befindet sich am Ausgang der Umwälzpumpen bzw. der Wärmeeinspeisung und wird Vorlaufdruck genannt. Anschließend nimmt der Druck durch vor allem reibungsbedingte Druckverluste hin zu den Wärmeverbrauchern ab. Die Wärmetauscher benötigen dort einen Mindestdifferenzdruck für die Wärmeentnahme und verringern somit den Druck von Vor- zu Rücklauf weiter [92, S. 59]. Im Rücklauf treten erneut Druckverluste auf bis zum niedrigsten Druck im System am Eingang der Umwälzpumpen bzw. der Wärmeeinspeisung, dem Rücklaufdruck. Abbildung 2.2 zeigt beispielhaft den Druckverlauf in einem Fernwärmenetz. Dort sind auch die Differenzdrücke der Wärmeverbraucher zu sehen und am Ende des Netzes der Netzschlechtepunkt, welcher den Wärmeverbraucher mit dem minimalen Differenzdruck zwischen Vor- und Rücklauf darstellt.

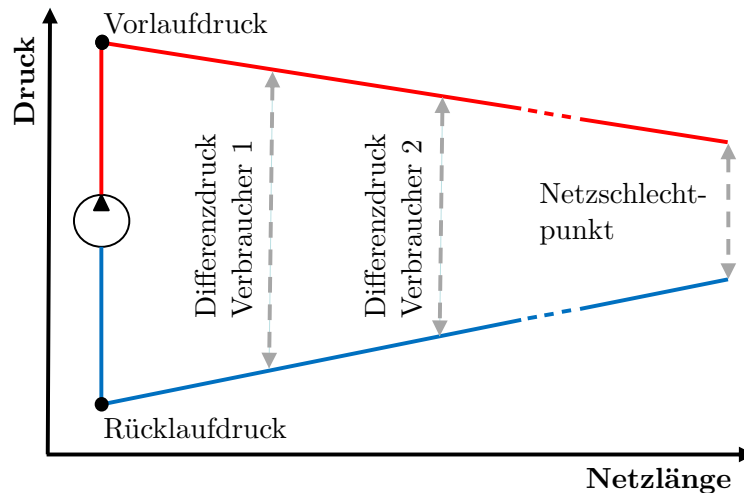


Abbildung 2.2.: Beispielhaftes Druckdiagramm eines Fernwärmenetzes nach [92, S. 59].

Der Differenzdruck zwischen Vor- und Rücklauf wird also durch die Umwälzpumpen bereitgestellt. Zusätzlich ist eine Druckhaltung notwendig, um das Druckniveau festzulegen. Dies erfolgt über Druckhaltungspumpen, welche mit einem Ausgleichsbehälter verbunden sind. In Fernwärmenetzen treten permanent geringe Wasserverluste durch nicht-ortbare Leckagen auf und auch Temperaturänderungen des Wassers führen zu Volumenänderungen im Leitungssystem. Fehlendes Wasservolumen kann durch die Druckhaltungspumpen aus dem Ausgleichsbehälter ins Fernwärmenetz nachgefüllt werden. Im Falle eines Überdrucks im System kann Wasser durch Überstromventile auch in den Ausgleichsbehälter zurückfließen. [92, S. 61]

Die für diese Arbeit relevanten Komponenten eines Fernwärmenetzes werden im Folgenden kurz vorgestellt. Dabei wird vordergründig auf die technischen Ausführungsarten der Komponenten und deren Funktionsweise eingegangen. Die physikalische Beschreibung und Berechnung der Komponenten wird in Kapitel 5 (*Thermo-hydraulische Modellierung von Fernwärmenetzen*) ausgeführt.

2.1.1. Wärmeerzeugungsanlagen

Wärmeerzeugungsanlagen speisen Wärmeenergie in Fernwärmenetze ein. Für diesen Zweck werden je nach Wärmebedarf und Ausdehnung eines Netzes eine oder mehrere Anlagen eingesetzt. Ein großer Teil der Wärme kann zum Beispiel von zentralen Heizkraftwerken (HKW) ins Netz gespeist werden, während bei Bedarf im System verteilte Heizwerke (HW) unterstützen. Die zentrale Versorgung kann von KWK-Anlagen übernommen werden, welche sowohl Strom als auch Wärme produzieren und gegenüber einer separaten Erzeugung höhere Wirkungsgrade aufweisen. In KWK-Anlagen wird

der Hochdruckdampf aus der Turbine nicht so weit entspannt wie bei der reinen Stromerzeugung, sondern die im Dampf enthaltene Wärme für Heißwasser oder als Prozessdampf verwendet. Durch diese frühere Entnahme geht ein Teil der im Dampf enthaltenen Energie für die Stromproduktion verloren, allerdings ist der Wirkungsgrad der Anlage durch die Nutzung der Wärme insgesamt höher. Der Energienutzungsgrad einer KWK-Anlage liegt beispielsweise bei Gas- und Dampfkraftwerken bei etwa 88 % verglichen mit 66 % bei getrennter Strom- und Wärmeerzeugung. Verluste lassen sich somit deutlich reduzieren. [69, S. 21-24]

Als Energiequellen werden fossile Energieträger wie Kohle, Gas oder Öl eingesetzt. Zusätzlich können Müllverbrennungsanlagen Wärme in die Netze einspeisen und auch regenerative Wärmequellen wie Geothermie, Biogas, Biomasse oder Solarthermie können verwendet werden. Wenn übermäßig viel Strom zum Beispiel durch Windenergie produziert wird, kommen auch Elektroerhitzer (*Power-To-Heat-Anlage*) zum Einsatz. In Industriegebieten kann entstehende Abwärme in lokale Dampfnetze oder ebenfalls in Fernwärmenetze eingespeist werden.

Speicher werden in Fernwärmenetzen eingesetzt, um eine flexiblere Versorgung zu erreichen. Prinzipiell kommen die drei Technologien der sensiblen, latenten und thermochemischen Wärmespeicherung infrage [92, S. 41]. Bei der sensiblen Wärmespeicherung wird erwärmtes Medium gespeichert. Die Energiedichte ist dabei im Vergleich am niedrigsten. Hier kann zum Beispiel Wasser in drucklosen Speichern bei unter 100 °C oder aufgrund des Dampfdrucks in bedrückten Speichern bei höheren Temperaturen gespeichert werden. Speicher können über- oder unterirdisch installiert und müssen mit zunehmender Speicherdauer besser von der Umgebung isoliert werden. Für Speicherung höherer oder niedriger Temperaturen können auch andere Medien wie zum Beispiel Thermoöl eingesetzt werden. Bei Latentwärmespeichern wird neben der Speicherung bei hohen Temperaturen auch die Energie für einen Phasenübergang gespeichert. Häufig werden Salze verflüssigt, welche bei der Entladung Wärme über die Abkühlung und Verfestigung abgeben. In thermochemischen Speichern werden chemische Reaktionsprodukte getrennt und gespeichert. Bei der Entladung findet eine exotherme Reaktion statt und Energie wird frei. Thermochemische und latente Speicher sind derzeit aufgrund des niedrigen Entwicklungsstands in Fernwärmenetzen kaum in der Anwendung. In gut isolierten Tanks, Latentwärmespeichern, thermochemischen oder unterirdischen Speichern wie zum Beispiel Aquifer- oder Kavernenspeichern kann Wärme über lange Zeit gelagert werden. Damit eignen sie sich auch als saisonale Speicher, welche die Unterschiede zwischen Sommer und Winter ausgleichen können, und sind somit für die zunehmende Verwendung regenerativer Energien interessant. Zum Beispiel kann Solar- oder Geothermiewärme, welche im Sommer überschüssig zur Verfügung steht, in saisonalen Speichern für den Winter gespeichert werden. [92, S. 42-47]

2.1.2. Pumpen

Pumpen werden für die Druckhaltung und Umwälzung des Mediums eingesetzt. Bei der Umwälzung kann ein Differenzdruck zwischen Ein- und Ausgang der Pumpe eingestellt oder ein bestimmtes Volumen gefördert werden. Die Pumpen sind in den Wärmeerzeugungsanlagen installiert oder werden dezentral im Netz verteilt als Druckerhöhungsanlagen (DEA) eingesetzt. Diese sind zum Beispiel in Fernwärmenetzen notwendig, in denen das Medium über weite Entfernungen mit hohen Druckverlusten gefördert wird. Umwälzpumpen oder DEAs können sowohl im Vor- als auch im Rücklauf eingesetzt werden. In der Regel werden in Pumpstationen drei Pumpen des gleichen Typs parallel geschaltet [69, S. 99]. Zwei Pumpen sind dann permanent im Betrieb, während die dritte Pumpe als Redundanz oder Reserve für hohe zu fördernde Wärmeströme dient. In Fernwärmenetzen werden Kreiselpumpen verschiedener Bauarten verwendet, welche stufenlos regelbar von Elektromotoren angetrieben werden [69, S. 100] [92, S. 51].

2.1.3. Ventile und Absperrarmaturen

Ventile werden eingesetzt, um als Regelventile den Durchfluss in Leitungen zu begrenzen oder komplett abzusperren. Vollständig geöffnete Ventile verhalten sich hydraulisch näherungsweise wie Rohrleitungen [92, S. 77]. Je nach Bauweise entsteht ein zusätzlicher Druckverlust durch in der Strömung befindliche Bauteile. Geschlossene Ventile verhindern den Durchfluss und werden eingesetzt, um bestimmte Versorgungsgebiete kurzzeitig für Wartungsarbeiten oder bei Leckagen abzusperren. Teilgeöffnete Ventile verursachen in der Strömung einen zusätzlichen Widerstand, welcher Druck und Durchfluss beeinflusst. So kann der Durchfluss und Druck in ein bestimmtes Gebiet begrenzt und ein Ventil zum Beispiel für die Regelung des Differenzdrucks eingesetzt werden. Des Weiteren haben Öffnungs- und Schließvorgänge dynamische Effekte, welche zu Druckspitzen führen können. Diese Vorgänge sollten somit nicht beliebig schnell erfolgen.

2.1.4. Rohrleitungen

Fernwärmenetze bestehen häufig aus zwei Teilnetzen: dem Vor- und Rücklauf. Ein solches Netz wird Zweileitersystem genannt. In einigen Fernwärmenetzen werden auch Dreileitersysteme eingesetzt. In diesen kann eine getrennte Versorgung mit Heißwasser bei konstanter und variabler Temperatur durch zwei Vorlaufleitungen erfolgen. Die Vorlaufleitung mit fixer Temperatur kann etwa für die Klimatisierung oder Versorgung mit Warmwasser eingesetzt werden [61, S. 9]. Auch ein Einsatz von Systemen mit nur einer Vorlaufleitung, aber zwei Rücklaufleitungen ist unter anderem bei stark unterschiedlichen Rücklauftemperaturen sinnvoll [92, S. 81]. Die Vorteile von zwei Vorlauf- und zwei Rücklaufleitungen lassen sich in Vierleiternetzen kombinieren. In dieser Arbeit werden nur Zwei- und Dreileitersysteme untersucht.

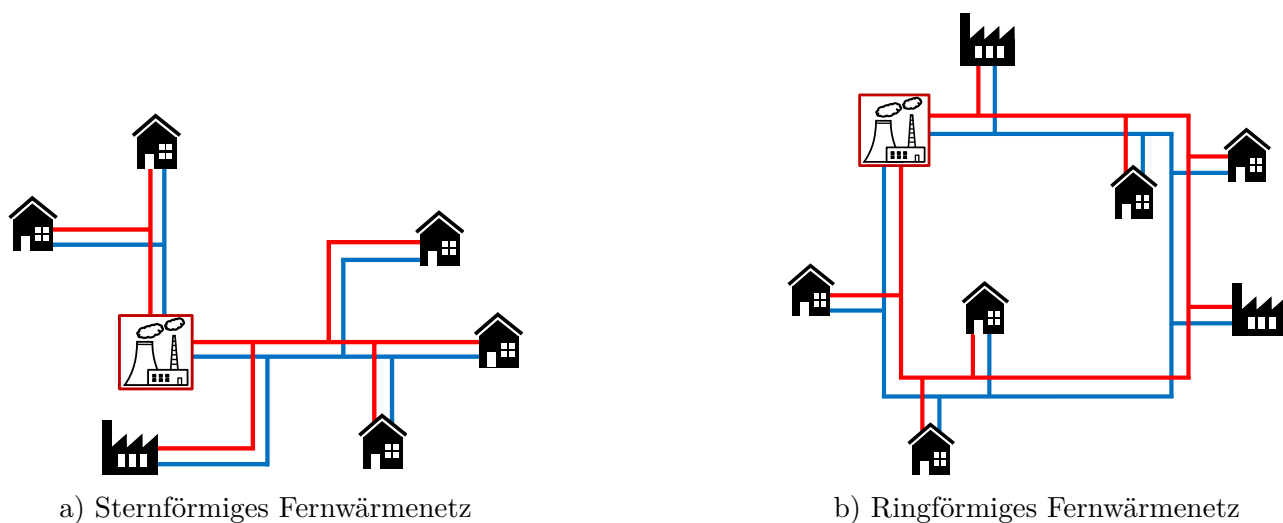


Abbildung 2.3.: Beispiele für Topologien von Fernwärmenetzen.

Fernwärmenetze können weiterhin in verschiedene Topologien kategorisiert werden. Je nach Aufbau des Rohrleitungssystems wird ein Netz als sternförmig, ringförmig oder vermascht bezeichnet [92, S. 81 f.]. Sternförmige oder auch strahlenförmig genannte Netze weisen die einfachste Topologie auf. Sie sind nicht vermascht und jeder Wärmetauscher im System ist genau auf einem Weg mit der Wärmeinspeisung verbunden. Existiert nur ein einspeisendes Kraftwerk im System, ist die Flussrichtung in den Leitungen stets gleichgerichtet. Ringförmige Netze bestehen aus einer ringförmigen Masche, an welche alle Wärmetauscher angeschlossen sind. Beispiele für stern- und ringförmige Netze sind in Abbildung 2.3 dargestellt. Die Münchner Fernwärmenetze sind historisch gewachsene vermaschte Netze wie zum Beispiel das Netz in Abbildung 2.4. In dieser Art Fernwärmenetz kann eine beliebige Anzahl

von Maschen vorhanden sein. Verbraucher können aus verschiedenen Richtungen mit Wärme versorgt werden. Vermaschte Netze machen die Versorgung flexibler und erhöhen durch die mögliche Versorgung der Abnehmer aus mehreren Richtungen die Versorgungssicherheit [92, S. 81]. Allerdings geht dies auf Kosten der Komplexität des Netzes. Im weiteren Verlauf dieser Arbeit werden vermaschte Netze behandelt.

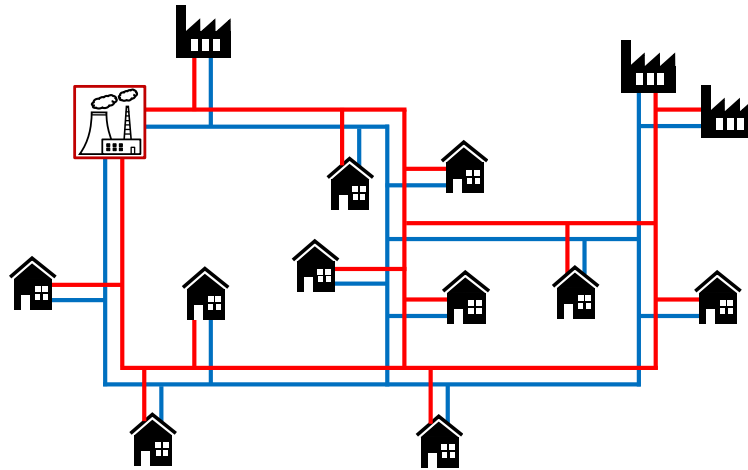


Abbildung 2.4.: Beispiel für ein maschenförmiges Fernwärmenetz.

In Fernwärmenetzen werden unterschiedliche Arten von Rohrleitungen abhängig von deren Anwendungszweck eingesetzt. Einflussfaktoren für die Entscheidung für einen Rohrtyp sind zum Beispiel das im Netz verwendete Medium (Wasser oder Dampf) sowie die Netzparameter wie Temperatur und Druck. Rohrleitungen bestehen aus einem inneren Rohr, auch Mediumrohr genannt, durch welches das Medium fließt, einer Wärmedämmung und einem Mantelrohr wie in Abbildung 2.5 dargestellt.

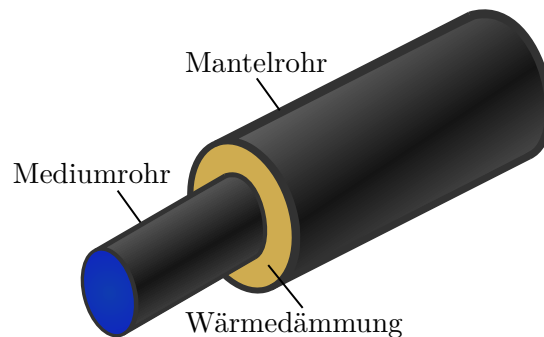


Abbildung 2.5.: Schematische Darstellung einer Rohrleitung mit Mediumrohr, Wärmedämmung und Mantelrohr nach [100, S. 78].

Für die Rohre wird als Material Stahl oder Kunststoff eingesetzt. Bei Stahlmantelrohren sind das innere und äußere Rohr aus Stahl gefertigt. Zusätzlich besteht die Wärmedämmung bei Stahlmantelrohren aus einer Wärmedämmung und einem evakuierten Raum. Die Rohre eignen sich für die höchsten Temperaturen bis 400°C und Drücke bis 64 bar sowie für die Verwendung bei hohen Außenbelastungen [69, S. 108] [92, S. 70]. Kunststoffmantelrohre werden aus einem inneren Rohr aus Stahl und einem äußeren Rohr aus Kunststoff gefertigt und halten aufgrund des verwendeten Wärmeschutzes aus Polyurethan im Dauerbetrieb nur Temperaturen von 130°C bis 140°C und Drücken bis 25 bar stand [69, S. 105] [92, S. 69]. Bei Vollkunststoffrohren, auch Kunststoffmediumrohre genannt, sind beide Rohre aus Kunststoff gefertigt und maximale Temperaturen von 95°C und maximale Drücke von 6 bar zulässig [92, S. 69]. Im Gegensatz zu Stahlrohren besteht bei Vollkunststoffrohren nicht die Gefahr der Korrosion. Allerdings eignen sie sich nicht für die in dieser Arbeit behandelten Fern-

wärmenetze mit hohen Temperaturen und werden nicht weiter behandelt.

Durch die verschiedenen Materialien haben die Rohre einen Einfluss auf die hydraulischen und thermischen Eigenschaften eines Fernwärmenetzes. Die Oberfläche des Mediumrohrs beeinflusst durch dessen Wandrauigkeit die hydraulischen Verluste. Die für die Rohre und Wärmedämmung verwendeten Materialien bestimmen wichtige Parameter des Wärmeverlusts. Da Rohre nicht perfekt von der Umgebung isoliert sind, entsteht beim Durchlaufen der Rohre ein Temperaturabfall und Wärme wird an die Umgebung abgegeben. Jede Materialschicht weist dabei eine bestimmte Wärmeleitfähigkeit auf. Zwischen den Materialien und Fluiden bestimmt der Wärmeübergangskoeffizient die Wärmeübertragung. Im Falle von Stahlmantelrohren kommt durch das zwischen Isolation und Mantelrohr vorhandene Vakuum noch die Übertragung durch Wärmestrahlung hinzu, welche durch den Emissions- und den Absorptionskoeffizienten der emittierenden und absorbierenden Schichten definiert ist. Auch die Verlegeart der Rohre über oder unter der Erde hat einen wesentlichen Einfluss auf den insgesamt für ein Rohr berechenbaren Wärmedurchgangskoeffizienten. Vor- und Nachteile von ober- oder unterirdische verlegten Rohrleitungen können zum Beispiel in [92, S. 84] nachgelesen werden.

2.1.5. Wärmetauscher

Wärmetauscher werden verwendet, um Wärme zwischen zwei Fluiden zu übertragen. In Wärmeerzeugungsanlagen übergeben Wärmetauscher Wärme von der Erzeugungsquelle an das Fluid im Rücklauf des Fernwärmenetzes. In Wärmeübergabestationen übergeben Wärmetauscher Wärme vom Vorlauf des Fernwärmenetzes an den Rücklauf der Abnehmer. Die Wasserkreisläufe der Erzeugung, des Fernwärmenetzes und der Abnehmer sind somit hydraulisch getrennt. Netze, bei welchen direkt das Medium des Netzes durch Heizkörper strömt [69, S. 5], werden hier nicht behandelt.

Die Wärmeübergabestationen haben eigene Regelungen, um den Durchfluss und somit den übertragenen Wärmestrom zu regeln. Für die korrekte Funktionsweise muss auf der Fernwärmenetzseite ein Mindestdifferenzdruck zur Verfügung gestellt werden [92, S. 59]. Die Wärmeübergabestationen regeln den Durchfluss so, dass die benötigte Wärmeleistung dem Medium entnommen und eine bestimmte Temperatur auf der Verbraucherseite erreicht wird. Außerdem muss am Ausgang der Station eine maximal zulässige Temperatur des Mediums im Rücklauf eingehalten werden [92, S. 152]. Niedrige Rücklauftemperaturen sind für die Effizienz des Fernwärmenetzes von Vorteil (siehe Kapitel 3.3.1).

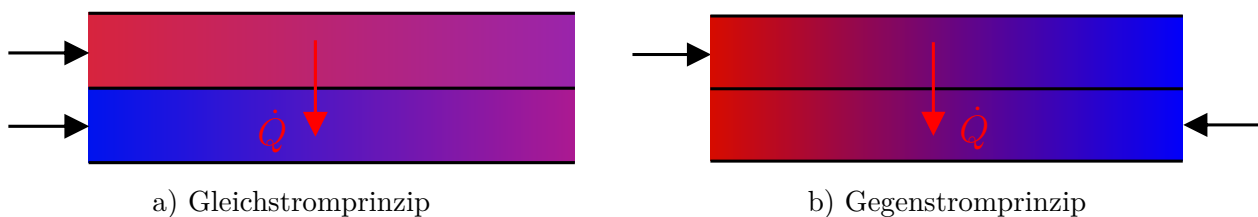


Abbildung 2.6.: Wärmetauscher nach dem Gleich- und Gegenstromprinzip nach [100].

Wärmetauscher werden in der Bauart als Plattenwärmetauscher oder in Rohrbauweisen verwendet. Zwei Medien werden jeweils in zwei oder mehreren Platten oder Rohren aneinander vorbeigeführt und tauschen Wärme aus. Dabei können unter anderem das Gleichstrom- oder Gegenstromprinzip genutzt werden. Beim Gleichstromprinzip fließen die Medien in die gleiche Richtung, beim Gegenstromprinzip entsprechend gegensätzlich. Beide Prinzipien sind in Abbildung 2.6 dargestellt. In Gegenstromwärmetauschern kann im Vergleich zu Gleichstromwärmetauschern bei gleicher Austauschfläche eine größere Wärmemenge vom warmen an das kalte Medium übertragen werden [128, S. 43].

2.2. Grundlagen der Optimierung

In dieser Arbeit soll die optimale Regelung verteilter Wärmeeinspeisung in Fernwärmenetzen untersucht werden. Daher werden hier Grundlagen der Optimierung und aktuell angewandter Verfahren vorgestellt.

2.2.1. Zielfunktion und Nebenbedingungen

Mathematisch kann eine Optimierung als die Minimierung oder Maximierung einer Funktion, der sogenannten Zielfunktion, beschrieben werden. Die Zielfunktion kann dabei weiterhin Nebenbedingungen unterliegen, welche ebenfalls durch Funktionen dargestellt werden und Restriktionen bzw. Grenzwerte darstellen. Das Optimierungsproblem lautet wie in [90, S. 3] und [95, S. 11] definiert:

$$\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) \quad (2.1)$$

unter Berücksichtigung von (u. B. v.)

$$\mathbf{c}(\mathbf{x}) = \mathbf{0}, \mathbf{c} \in \mathbb{R}^m, \quad (2.2)$$

$$\mathbf{h}(\mathbf{x}) \leq \mathbf{0}, \mathbf{h} \in \mathbb{R}^q. \quad (2.3)$$

Das Ziel einer Optimierung ist die Minimierung der skalaren Zielfunktion $f(\mathbf{x})$. Der Dimension n des Variablenvektors \mathbf{x} sind keine Grenzen gesetzt. Bei der Optimierung sollen die Gleichungsnebenbedingungen $\mathbf{c}(\mathbf{x})$ und die Ungleichungsnebenbedingungen $\mathbf{h}(\mathbf{x})$ eingehalten werden. Für eine Optimierung muss die Dimension m kleiner als n sein, da \mathbf{x} sich sonst aus den Gleichungsnebenbedingungen ergibt. Der Dimension q der Ungleichungsnebenbedingungen sind keine Grenzen gesetzt. [95, S. 11]

2.2.2. Klassifikation von Optimierungsproblemen

Optimierungsprobleme zeichnen sich durch verschiedene Charakteristika aus, welche hier grundlegend vorgestellt werden. Zunächst kann zwischen Problemen mit und ohne Nebenbedingungen unterschieden werden. In dieser Arbeit werden Optimierungen mit Nebenbedingungen durchgeführt. Des Weiteren kann nach den verwendeten Zielfunktionen und Nebenbedingungen klassifiziert werden. Besteht ein Problem aus einer linearen Zielfunktion mit linearen Gleichungs- und Ungleichungsnebenbedingungen, handelt es sich um eine lineare Optimierung, auch lineare Programmierung oder lineares Programm genannt. Sind entweder Zielfunktionen oder Nebenbedingungen nicht-linear, wird das Problem als nicht-lineare Optimierung beschrieben. Eine weitere Klasse von Optimierungsproblemen und Unterklasse der nicht-linearen Optimierung ist die quadratische Programmierung. Diese besteht aus der Optimierung einer quadratischen Zielfunktion mit linearen Nebenbedingungen [90, S. 448]. Für die jeweiligen Optimierungsprobleme gibt es Methoden, welche je nach spezifischem Anwendungsfall verwendet werden können. Methoden der einfacher lösbarer linearen und quadratischen Optimierung können auch für nicht-lineare Optimierungsprobleme adaptiert werden [90, S. 101].

Weiterhin kann zwischen diskreten und stetigen Optimierungen unterschieden werden [90, S. 5]. Bei diskreten Optimierungen werden binäre oder ganzzahlige Variablen verwendet. Der Einsatz von binären Variablen kann zum Beispiel für die Bestimmung des An- und Aus-Zustands von Heizwerken verwendet werden. Probleme dieser Art werden, wie später in Kapitel 3.3.2 beschrieben, oft als gemischt ganzzahlige lineare Probleme formuliert, auch *Mixed-Integer Linear Programming* (MILP) genannt. Bei stetigen Optimierungen sind alle Variablen und entsprechend auch die Zielfunktion und Restriktionen stetig. Diese Art von Problemen sind in der Regel einfacher lösbar, da keine Sprünge in

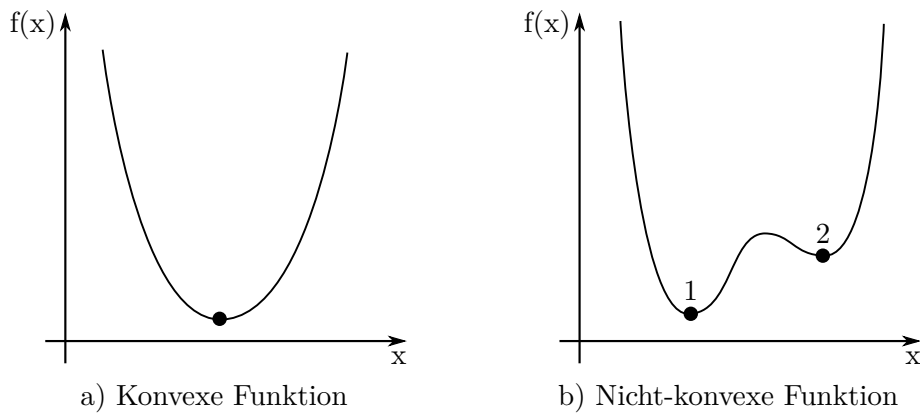


Abbildung 2.7.: Konvexe Funktion mit markiertem globalem Minimum und nicht-konvexe Funktion mit einem globalen (1) sowie einem lokalen (2) Minimum, nach [95, S. 15].

den Werten der Zielfunktion und Nebenbedingungen entstehen [90, S. 5]. Bei der Optimierung der Erzeugungsanlagen von Fernwärmenetzen mit An- und Aus-Zuständen und nicht-linearen Modellen als Restriktionen handelt es sich um ein gemischt ganzzahliges nicht-lineares Problem, auch *Mixed-Integer Non-Linear Programming* (MINLP) genannt.

Wichtig ist ebenfalls die Unterscheidung zwischen der globalen und lokalen Optimierung. Bei der lokalen Optimierung wird ein lokales Optimum gesucht, an welchem die Restriktionen erfüllt werden und der Wert der Zielfunktion kleiner ist als der Wert aller umliegenden Punkte [90, S. 7]. Oft sind lokale Optima ausreichend für eine Lösung eines Problems. Handelt es sich um ein konvexes oder lineares Optimierungsproblem, ist ein lokales Optimum auch das Globale [90, S. 7]. Bei nicht-konvexen Problemen ist dies nicht der Fall und das globale Optimum lässt sich oft nur schwer bestimmen. Die Schwierigkeit lässt sich anhand des Beispiels einer konvexen und einer nicht-konvexen Funktion in Abbildung 2.7 darstellen. Die konvexe Funktion hat nur ein globales Minimum, während die nicht-konvexe Funktion über zwei lokale Minima verfügt, von welchen nur eines das globale Minimum ist. In [95] sind die mathematischen Beschreibungen zum Beispiel für konvexe Funktionen und genauere Definitionen der Optimierungsprobleme beschrieben, welche hier nicht weiter ausgeführt werden sollen.

Zuletzt kann zwischen der stochastischen und deterministischen Optimierung unterschieden werden [90, S. 7]. Bei deterministischen Optimierungen sind die Gleichungen für die Zielfunktion und Restriktionen vollständig bekannt. Bei einigen Problemen ist dies nicht der Fall und Parameter des verwendeten Modells sind nicht bekannt oder unsichere Eingangsdaten sind vorhanden. Wenn allerdings Erwartungswerte bekannt sind oder berechnet werden können, können diese anschließend in stochastischen Optimierungen berücksichtigt werden. In der robusten Optimierung müssen einige Restriktionen so für alle möglichen Werte einer Variablen eingehalten werden. Stochastische Optimierungen werden häufig durch die Formulierung mehrerer deterministischer Optimierungsprobleme gelöst [90, S. 7]. Nicht zu verwechseln ist die stochastische Optimierung mit stochastischen Verfahren für die Optimierung, welche in Kapitel 2.2.6 behandelt werden und auf stochastischer Änderung der Variablen während des Optimierungsprozesses basieren.

2.2.3. Optimalitätsbedingungen

Hier werden die notwendigen und hinreichenden Bedingungen für ein lokales Minimum beschrieben. Für ein Minimum einer Funktion mit einer Variablen und ohne Nebenbedingungen ist eine notwendige Bedingung erster Ordnung, dass die erste Ableitung am Minimum x^* gleich 0 wird [95, S. 22]:

$$f'(x^*) = 0 \quad (2.4)$$

Als notwendige Bedingung zweiter Ordnung gilt [95, S. 22]:

$$f'(x^*) = 0 \quad \text{und} \quad f''(x^*) \geq 0 \quad (2.5)$$

Als letztes folgt die hinreichende Bedingung, welche besagt, dass die zweite Ableitung größer 0 sein muss [95, S. 22]:

$$f'(x^*) = 0 \quad \text{und} \quad f''(x^*) > 0 \quad (2.6)$$

Bei einer Optimierung müssen die hinreichenden Bedingungen nicht unbedingt erfüllt werden [90, S. 16]. Für den Mehrgrößenfall gelten folgende erweiterte Bedingungen für ein Minimum \mathbf{x}^* [95, S. 39]:

$$\text{Notwendige Bedingung 1. Ordnung: } \nabla f(\mathbf{x}^*) = \mathbf{0} \quad (2.7)$$

$$\text{Notwendige Bedingung 2. Ordnung: } \nabla f(\mathbf{x}^*) = \mathbf{0} \quad \text{und} \quad \nabla^2 f(\mathbf{x}^*) \succeq \mathbf{0} \quad (2.8)$$

$$\text{Hinreichende Bedingung: } \nabla f(\mathbf{x}^*) = \mathbf{0} \quad \text{und} \quad \nabla^2 f(\mathbf{x}^*) \succ \mathbf{0} \quad (2.9)$$

Bei Funktionen mit mehreren Variablen muss der Gradient $\nabla f(\mathbf{x}^*)$ gleich null und die *Hesse*-Matrix $\nabla^2 f(\mathbf{x}^*)$ positiv definit sein, also nur Eigenwerte größer 0 besitzen. Ein lokales Minimum liegt so an einem Punkt, von welchem die Zielfunktion in jede Bewegungsrichtung nur ansteigt. Der Gradient und die *Hesse*-Matrix der Zielfunktion sind folgendermaßen definiert:

$$\nabla f(\mathbf{x}^*) = \begin{bmatrix} \frac{\partial f(\mathbf{x}^*)}{\partial x_1^*} \\ \vdots \\ \frac{\partial f(\mathbf{x}^*)}{\partial x_n^*} \end{bmatrix} \quad (2.10)$$

$$\nabla^2 f(\mathbf{x}^*) = \begin{bmatrix} \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_1^{*2}} & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_1^* \partial x_2^*} & \cdots & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_1^* \partial x_n^*} \\ \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_2^* \partial x_1^*} & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_2^{*2}} & \cdots & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_2^* \partial x_n^*} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_n^* \partial x_1^*} & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_n^* \partial x_2^*} & \cdots & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_n^{*2}} \end{bmatrix} \quad (2.11)$$

Die bisherigen Bedingungen beschreiben das Minimum der Zielfunktion $f(\mathbf{x}^*)$ des Optimierungsproblems. Allerdings werden keine Gleichungs- und Ungleichungsnebenbedingungen berücksichtigt, welche in dieser Arbeit notwendig sind. Für diesen Fall wird zunächst eine Funktion definiert, welche sowohl die Zielfunktion als auch Gleichungs- und Ungleichungsnebenbedingungen enthält. An diese sogenannte *Lagrange*-Funktion können anschließend Bedingungen gestellt werden. Die skalare Lagrange-Funktion \mathcal{L} kann folgendermaßen mit dem Vektor der *Lagrange*-Multiplikatoren $\boldsymbol{\lambda} \in \mathbb{R}^m$ und dem Vektor der *Kuhn-Tucker*-Multiplikatoren $\boldsymbol{\mu} \in \mathbb{R}^q$ formuliert werden [95, S. 87]:

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{c}(\mathbf{x}) + \boldsymbol{\mu}^T \mathbf{h}(\mathbf{x}) \quad (2.12)$$

Wie bereits in den vorangegangenen Beispielen gilt, dass die notwendige Bedingung 1. Ordnung eine Ableitung bzw. einen Gradienten von $\mathbf{0}$ der Zielfunktion am lokalen Minimum \mathbf{x}^* voraussetzt. Hier werden nun Anforderungen an die Gradienten der *Lagrange*-Funktion gestellt. Es existieren die Multiplikatoren $\boldsymbol{\lambda}^*$ und $\boldsymbol{\mu}^*$, so dass [95, S. 88]:

$$\nabla_x \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = \nabla_x f(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T} \nabla_x \mathbf{c}(\mathbf{x}^*) + \boldsymbol{\mu}^{*T} \nabla_x \mathbf{h}(\mathbf{x}^*) = \mathbf{0} \quad (2.13)$$

$$\nabla_{\boldsymbol{\lambda}} \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = \mathbf{c}(\mathbf{x}^*) = \mathbf{0} \quad (2.14)$$

$$\mathbf{h}(\mathbf{x}^*) \leq \mathbf{0} \quad (2.15)$$

$$\mathbf{h}(\mathbf{x}^*)^T \boldsymbol{\mu}^* = \mathbf{0} \quad (2.16)$$

$$\boldsymbol{\mu}^* \geq \mathbf{0} \quad (2.17)$$

Zunächst muss der Gradient der *Lagrange*-Funktion gemäß Gleichung (2.13) null werden. In diesem Fall wurde ein lokales Minimum gefunden, von welchem aus nur Bewegungsrichtungen existieren, in welche entweder die Zielfunktion wächst oder die Nebenbedingungen verletzt werden. Zudem müssen nach (2.14) die Gleichungs- und nach (2.15) die Ungleichungsnebenbedingungen erfüllt sein. Die Bedingungen (2.16) und (2.17) schränken die Problemstellung weiter ein und helfen bei der Definition der Optimierungsrichtung. Hier wird zwischen aktiven und inaktiven Nebenbedingungen unterschieden. Für eine aktive Nebenbedingung gilt $h_i(\mathbf{x}) = 0$, für eine inaktive Nebenbedingung $h_i(\mathbf{x}) < 0$. Durch die Definition von (2.16) und (2.17) werden inaktive Nebenbedingungen in einem Optimum nicht berücksichtigt. Des Weiteren kann bei erfüllten Bedingungen (2.13) und (2.16) und nicht erfüllter Bedingung (2.17) eine Richtung definiert werden, welche die Lagrange-Funktion weiter verkleinert [95, S. 88 f.]. Die notwendigen Bedingungen 1. Ordnung werden auch *Kuhn-Tucker*- [95, S. 87] bzw. *Karush-Kuhn-Tucker*-Bedingungen [90, S. 321] genannt.

Die notwendige Bedingung 2. Ordnung (2.18) und hinreichende Bedingung (2.19) für ein lokales Optimum können nun wieder über die positive Semi-Definitheit, bzw. positive Definitheit der *Hesse*-Matrix, in diesem Falle der *Lagrange*-Funktion gestellt werden [95, S. 92]. In beiden Fällen müssen auch die notwendigen Bedingungen 1. Ordnung (2.13) bis (2.17) erfüllt sein.

$$\nabla_{xx}^2 \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) \succeq \mathbf{0} \quad (2.18)$$

$$\nabla_{xx}^2 \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) \succ \mathbf{0} \quad (2.19)$$

Nachdem die Bedingungen für Minima von Optimierungsproblemen ohne und mit Nebenbedingungen definiert wurden, werden im Folgenden einige Optimierungsalgorithmen vorgestellt, welche später in Optimierungen für Ansätze der Systemidentifikation, des maschinellen Lernens oder die Optimierung im Rahmen einer MPC verwendet werden. Gängige Methoden werden zum Beispiel in [90] und [95] detailliert beschrieben. Einfache Probleme können analytisch gelöst werden. Ist dies nicht der Fall, werden numerische Verfahren für die Bestimmung lokaler Optima verwendet. Einige Verfahren werden zunächst theoretisch vorgestellt. Die in dieser Arbeit verwendeten Umsetzungen werden anschließend am Ende von Kapitel 2.2.5 beschrieben.

2.2.4. Lokale Optimierung ohne Nebenbedingungen

Zunächst werden die Grundlagen der numerischen Verfahren für lokale Optimierungen ohne Nebenbedingungen beschrieben. Die in dieser Arbeit behandelten Optimierungsprobleme sind zwar solche mit Gleichungs- und Ungleichungsnebenbedingungen, allerdings werden die beschriebenen Verfahren auch dort mit veränderten Zielfunktionen und erweiterten Ansätzen eingesetzt. Lokale Optimierungsalgorithmen ohne Nebenbedingungen werden zum Beispiel in [90, S. 10 ff.] und [95, S. 37 ff.] vorgestellt. In diesem Kapitel wird ein Überblick der vorhandenen Algorithmen nach [90, S. 18 ff.] gegeben.

Zunächst muss ein Startpunkt \mathbf{x}_0 angegeben oder vom Algorithmus festgelegt werden. Anschließend werden von einem Algorithmus Iterationen $\{\mathbf{x}_k\}_{k=0}^{\text{inf}}$ gebildet, bis ein lokales Minimum mit ausreichender Genauigkeit gefunden oder Abbruchkriterien erfüllt werden, welche zum Beispiel durch fehlenden Fortschritt bei der Optimierung festgelegt sind. An einem Punkt \mathbf{x}_k sucht ein Algorithmus einen Punkt \mathbf{x}_{k+1} , an welchem die Zielfunktion $f(\mathbf{x})$ des Problems kleiner wird. Die am häufigsten verwendeten Algorithmen, welche die Richtung und Weite des nächsten Schritts definieren, sind die Algorithmen *Line-Search* und *Trust-Region* [90, S. 18].

Line-Search

Line-Search sucht entlang einer Richtung \mathbf{p}_k die Schrittweite $\alpha > 0$, welche die Funktion

$$f(\mathbf{x}_k + \alpha \mathbf{p}_k) \tag{2.20}$$

minimiert [90, S. 19]. Die Suchrichtung kann zum Beispiel durch das Gradientenverfahren bestimmt werden, auch *gradient descent* oder *method of steepest descent* genannt. An dem Punkt k wird dabei der Gradient der Zielfunktion berechnet und anschließend der negative Gradient $-\nabla f(\mathbf{x})$ als Suchrichtung festgelegt [90, S. 20 f.]. Das Verfahren kann bei komplexen Optimierungsproblemen zu langen Rechenzeiten führen. Andere Verfahren zur Bestimmung der Suchrichtung sind daher zum Beispiel die *Newton-Verfahren*. Bei diesen basiert die Suchrichtung auf der Annäherung der Zielfunktion als Modellfunktion in Form eines *Taylorpolynoms* zweiter Ordnung [90, S. 22]:

$$f(\mathbf{x}_k + \mathbf{p}) \approx f(\mathbf{x}_k) + \mathbf{p}^T \nabla f(\mathbf{x}_k) + \frac{1}{2} \mathbf{p}^T \nabla^2 f(\mathbf{x}_k) \mathbf{p} \stackrel{\text{def}}{=} m_k(\mathbf{p}) \tag{2.21}$$

Diese quadratische Modellfunktion $m_k(\mathbf{p})$ wird durch die *Newton-Suchrichtung* \mathbf{p}_k^N minimiert, welche durch Nullsetzen der Ableitung der Modellfunktion gefunden wird [90, S. 22]:

$$\mathbf{p}_k^N = -(\nabla^2 f(\mathbf{x}_k))^{-1} \nabla f(\mathbf{x}_k) \tag{2.22}$$

Ein Vorteil von *Newton-Methoden* ist die hohe Konvergenzgeschwindigkeit bei gut gewählten Startwerten. Außerdem kann die Schrittweite α zu 1 gewählt und muss nur geändert werden, falls mit dieser Schrittweite keine ausreichende Reduktion der Zielfunktion gefunden wird. Ein Nachteil der *Newton-Suchrichtung* ist, dass die *Hesse-Matrix* der Zielfunktion benötigt wird. Kann die *Hesse-Matrix* nicht explizit formuliert werden, kann sie zum Beispiel durch *Finite-Differenzen-Methoden* angenähert werden. Gleiches gilt für den Gradienten, welcher sowohl für die *Newton-Suchrichtung* als auch für das *Gradientenverfahren* benötigt wird. [90, S. 22 ff.]

Eine Alternative zu den beiden bisher genannten Ansätzen bieten die sogenannten *Quasi-Newton-Verfahren*, welche die *Hesse-Matrix* der Zielfunktion nur annähern und in jedem Rechenschritt wieder aktualisieren. Die Annäherung \mathbf{B}_k , welche hier nicht näher beschrieben wird, wird anschließend als Ersatz für die *Hesse-Matrix* der Zielfunktion in (2.22) eingesetzt [90, S. 24]:

$$\mathbf{p}_k = -\mathbf{B}_k^{-1} \nabla f(\mathbf{x}_k) \tag{2.23}$$

Ein bekannter Ansatz ist zum Beispiel der *BFGS-Algorithmus* nach Broyden, Fletcher, Goldfarb und Shanno [90, S. 24 f.]. Für die Definition der Suchrichtung von *Line-Search-Verfahren* gibt es weitere Ansätze, wie zum Beispiel *konjugierte Gradientenverfahren*.

Trust-Region

Die Alternative zum *Line-Search-Ansatz* *Trust-Region* definiert eine Modellfunktion m_k , welche sich um den Punkt \mathbf{x}_k ähnlich der Zielfunktion $f(\mathbf{x})$ des Problems verhält. Anschließend wird der Schritt \mathbf{p} durch die Minimierung von (2.24) definiert [90, S. 19].

$$m_k(\mathbf{x}_k + \mathbf{p}) \tag{2.24}$$

\mathbf{p} muss dabei innerhalb der *Trust-Region*, einer maximalen Schrittweite Δk liegen. Falls die Zielfunktion durch das Resultat nicht ausreichend verkleinert werden kann, wird die *Trust-Region* reduziert und das Problem erneut gelöst. Im Gegensatz zu dem *Line-Search-Verfahren* wird also zuerst die Schrittweite und anschließend die Richtung festgelegt. Die Modellfunktion wird in der Regel als quadratische

Funktion definiert [90, S. 19]:

$$m_k(\mathbf{x}_k + \mathbf{p}) = f(\mathbf{x}_k) + \mathbf{p}^T \nabla f(\mathbf{x}_k) + \frac{1}{2} \mathbf{p}^T \mathbf{B}_k \mathbf{p} \quad (2.25)$$

\mathbf{B}_k ist hier, wie bei den bereits vorgestellten *Newton*-Verfahren, die *Hesse*-Matrix der Zielfunktion oder eine Annäherung dieser und kann auch gemäß eines Gradientenverfahrens als $\mathbf{0}$ definiert werden.

2.2.5. Lokale Optimierung mit Nebenbedingungen

Für lineare Funktionen mit Nebenbedingungen existieren Methoden des *Linear Programming* (LP), welche in einigen Optimierungen für Fernwärmenetze in Kapitel 3.3.2 verwendet werden. Hier soll allerdings nicht auf diese eingegangen werden, da in der vorliegenden Arbeit hauptsächlich nicht-lineare Probleme behandelt und deshalb in diesem Abschnitt einige der Algorithmen für die nicht-lineare Optimierung mit Nebenbedingungen kurz erläutert werden. Für quadratische Funktionen steht zum Beispiel *Quadratic Programming* (QP) zur Verfügung. Andere Algorithmen sind *Sequential Quadratic Programming* (SQP) und *Interior-Point*.

Quadratic Programming

Bei der quadratischen Programmierung handelt es sich um ein Verfahren zur Lösung von Problemen mit quadratischen Zielfunktionen und linearen Nebenbedingungen [95, S. 111]. Für die Lösung der in dieser Arbeit auftretenden nicht-linearen Probleme ist dieses Verfahren also nicht direkt geeignet, wird allerdings beispielsweise in den SQP Verfahren als Subproblem verwendet [90, S. 448]. Das Optimierungsproblem ist hier [95, S. 111]:

$$\min \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{p}^T \mathbf{x} \quad (2.26)$$

mit $\mathbf{x} \in \mathbb{R}^n$ und $\mathbf{Q} \in \mathbb{R}^{n \times n}$. \mathbf{Q} muss dabei symmetrisch und positiv definit sein. Die Gleichungs- und Ungleichungsnebenbedingungen lauten [95, S. 111]:

$$\mathbf{A} \mathbf{x} = \mathbf{a} \quad (2.27)$$

$$\mathbf{B} \mathbf{x} \leq \mathbf{b} \quad (2.28)$$

mit $\mathbf{A} \in \mathbb{R}^{m \times n}$ und $\mathbf{B} \in \mathbb{R}^{q \times n}$, mit $m < n$ und beliebigem q . Das Problem kann zum Beispiel durch *active-set*-Verfahren gelöst werden. Zu Anfang der Optimierung muss dabei ein zulässiger Punkt \mathbf{x}_0 mit aktiven Ungleichungsnebenbedingungen $\mathbf{B}_a \mathbf{x}_0$ bekannt sein. Die Ungleichungsnebenbedingungen (2.28) reduzieren sich damit zu [95, S. 111]:

$$\mathbf{B}_a \mathbf{x} = \mathbf{b}_a \quad (2.29)$$

Anschließend kann für dieses Problem eine zu minimierende *Lagrange*-Funktion entsprechend Kapitel 2.2.3 formuliert werden [95, S. 112]:

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{p}^T \mathbf{x} + \boldsymbol{\lambda}^T (\mathbf{A} \mathbf{x} - \mathbf{a}) + \boldsymbol{\mu}^T (\mathbf{B}_a \mathbf{x} - \mathbf{b}_a) \quad (2.30)$$

Die notwendigen Bedingungen 1. Ordnung sind linear [95, S. 112]:

$$\nabla_{\mathbf{x}} \mathcal{L} = \mathbf{Q} \mathbf{x} + \mathbf{p} + \mathbf{A}^T \boldsymbol{\lambda} + \mathbf{B}_a^T \boldsymbol{\mu} = \mathbf{0} \quad (2.31)$$

$$\nabla_{\boldsymbol{\lambda}} \mathcal{L} = \mathbf{A} \mathbf{x} - \mathbf{a} = \mathbf{0} \quad (2.32)$$

$$\nabla_{\boldsymbol{\mu}} \mathcal{L} = \mathbf{B}_a \mathbf{x} - \mathbf{b}_a = \mathbf{0} \quad (2.33)$$

In *active-set*-Verfahren nach [95, S. 112 f.] und [90, S. 467 ff.] wird das quadratische Problem iterativ gelöst. In jedem Schritt werden eine Menge aktiver Ungleichungen bestimmt und das Gleichungssystem neu gelöst, solange bis ein Minimum gefunden wurde. Dabei werden jeweils die Suchrichtung und anschließend die Schrittweite bestimmt. Auch andere Ansätze wie die *Interior-Point*-Methode können für die Lösung in jedem Schritt verwendet werden [90, S. 480].

Sequential Quadratic Programming

Beim Ansatz SQP wird das Problem der nicht-linearen Minimierung mit Nebenbedingungen gelöst. Als Bedingung für ein Optimum gelten also (2.13) bis (2.19). Die *Lagrange*-Funktion wird in jedem Schritt als quadratische sowie die Nebenbedingungen als lineare Funktionen approximiert. Das folgende so entstehende Problem kann in jedem Schritt mittels quadratischer Programmierung, wie vorab beschrieben, gelöst werden [90, S. 533], vgl. [95, S. 114 ff.]:

$$\min_{\mathbf{p}} f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T \mathbf{p} + \frac{1}{2} \mathbf{p}^T \nabla_{xx}^2 \mathcal{L}_k \mathbf{p} \quad (2.34)$$

$$\text{u. B. v. } \nabla \mathbf{c}(\mathbf{x}_k)^T \mathbf{p} + \mathbf{c}(\mathbf{x}_k) = \mathbf{0}, \quad (2.35)$$

$$\nabla \mathbf{h}(\mathbf{x}_k)^T \mathbf{p} + \mathbf{h}(\mathbf{x}_k) \leq \mathbf{0}. \quad (2.36)$$

Wie der Name des Verfahrens sagt, werden sequenziell Probleme der quadratischen Programmierung gelöst. Dabei können die im vorherigen Abschnitt beschriebenen *active-set*-Methoden [90, S. 529] angewandt werden. In jedem Schritt müssen aktive Ungleichungsnebenbedingungen vorhanden sein, welche zu Gleichungsnebenbedingungen umformuliert und anschließend für die Lösung eines Problems ohne Ungleichungsnebenbedingungen verwendet werden. Prinzipiell befindet sich der Algorithmus also jederzeit an einem Punkt, an welchem Ungleichungsnebenbedingungen aktiv sind. Dieser äußere zulässige Rand des Lösungsbereichs wird bis ins Minimum verfolgt [95, S. 117].

Nach [90, S. 530], [71, S. 10] und [95, S. 113] kann SQP als Newton-Verfahren für die Lösung beschränkter nicht-linearer Gleichungssysteme bzw. der *Karush-Kuhn-Tucker*-Bedingungen aus Kapitel 2.2.3 gesehen werden. Des Weiteren gibt es auch *Trust-Region*-Ansätze für SQP. Da ein großer Teil des Rechenaufwands durch die Berechnung der Hesse-Matrix entsteht, kann diese wieder durch den Quasi-Newton-Ansatz angenähert werden.

Interior-Point-Verfahren

Das *Interior-Point*-Verfahren wurde zunächst für die Lineare Programmierung eingesetzt und später auch für die Lösung nicht-linearer Optimierungsprobleme. Nach Papageorgiou et al. [95, S. 117] sind *Interior-Point*-Verfahren zusammen mit SQP die am häufigsten eingesetzten Algorithmen für die Optimierung von nicht-linearen Problemen mit Nebenbedingungen. Im Gegensatz zu SQP wird beim *Interior-Point*-Verfahren nicht ausgehend von aktiven Ungleichungsnebenbedingungen optimiert. Hier wird aus dem zulässigen Bereich vorgegangen, bis ein Minimum erreicht wird. In jedem Schritt müssen die Ungleichungsnebenbedingungen strikt erfüllt werden, also $\mathbf{h}(\mathbf{x}) < \mathbf{0}$. Der sogenannte Barriere-Parameter bestimmt dabei den Abstand vom Rand des Lösungsbereichs. Somit kann das Optimierungsproblem unter der Verwendung der Schlupfvariablen \mathbf{s} neu definiert werden [95, S. 118]:

$$\min_{\mathbf{x}, \mathbf{s}} f(\mathbf{x}) \quad (2.37)$$

$$\text{u. B. v. } \mathbf{c}(\mathbf{x}) = \mathbf{0}, \quad (2.38)$$

$$\mathbf{h}(\mathbf{x}) + \mathbf{s} = \mathbf{0} \quad (2.39)$$

$$\mathbf{s} \geq \mathbf{0}. \quad (2.40)$$

Für dieses Problem werden wieder die *Lagrange*-Funktion sowie die notwendigen Bedingungen 1. Ordnung bestimmt, was hier nicht näher ausgeführt wird. Das aus den notwendigen Bedingungen entstehende nicht-lineare Gleichungssystem kann zum Beispiel durch das Newton-Verfahren gelöst werden, um eine Suchrichtung zu erhalten. Anschließend werden die Schrittweite bestimmt und das Problem in dem erreichten Punkt erneut gelöst. Der Algorithmus wird wiederholt, bis ein Minimum gefunden wurde. Statt dieser Vorgehensweise nach dem *Line-Search*-Verfahren kann auch wieder der *Trust-Region*-Ansatz verwendet werden. [95, S. 118 ff.]

Implementierungen der Verfahren

Im Folgenden werden kurz einige der Optimierer aufgeführt, welche in den späteren Teilen der Arbeit verwendet und getestet werden. Zum Beispiel ist der Optimierer *We Optimize Really Huge Problems* (WORHP) [22] zu nennen. Dieser Algorithmus basiert auf dem SQP-Ansatz. Die Lösung der approximierten quadratischen Subprobleme mittels des *active-set*-Verfahrens wird allerdings bei großen Problemen sehr komplex [95, S. 117 f.]. Stattdessen wird bei WORHP ein *Interior-Point*-Algorithmus verwendet, um die Suchrichtung zu finden und anschließend die Schrittweite durch *Line-Search* ermittelt. Die Ableitungen erster und zweiter Ordnung sind bei der Optimierung nötig, können allerdings auch von WORHP selbst durch Finite-Differenzen-Verfahren oder *Quasi-Newton*-Methoden angenähert werden. WORHP kann zum Beispiel in *MATLAB*[®] oder *C++* eingebunden werden.

Des Weiteren wird der Optimierer *fmincon* von *MATLAB*[®] verwendet [122]. Bei diesem Optimierer stehen einige unterschiedliche Algorithmen für die Lösung nicht-linearer Probleme mit Nebenbedingungen zur Auswahl. Zum Beispiel kann ein SQP-Algorithmus gewählt werden, welcher allerdings das quadratische Subproblem nach dem *active-set*-Verfahren löst. Zusätzlich sind ein *Trust-Region*- sowie ein *Interior-Point*-Ansatz implementiert.

In Python wird der Optimierer *minimize* eingesetzt. Die hier angewandte Methode *Sequential Least Squares Programming* (SLSQP) ist ein SQP-Algorithmus, welcher in der *Python* Bibliothek *scipy* [124] implementiert und in [71] beschrieben wird. Die Suchrichtung wird durch ein Subproblem der quadratischen Programmierung, welches in ein lineares *least-squares*-Problem (Methode der kleinsten Quadrate) transformiert wird, bestimmt [71, S. 20].

2.2.6. Globale Optimierung

Die in den letzten Kapiteln vorgestellten Algorithmen finden bei vielen Optimierungsproblemen nur lokale Optima. Bei Problemen mit vielen lokalen Optima entspricht ein gefundenes lokales Optimum nicht mehr unbedingt einer guten Lösung. Die gefundene Lösung hängt auch besonders von den für die Optimierung gewählten Startwerten ab. Für solche Probleme existieren weitere Optimierungsverfahren, welche in der Lage sind, globale Optima oder zumindest mehrere lokale Optima zu finden, aus welchen dann entsprechend das beste ausgewählt wird. Hier kommen stochastische Verfahren zum Einsatz, welche unter anderem in [95, S. 60 ff.] erläutert werden. Bei diesen Verfahren werden gezielt Zufallsvariablen eingesetzt und Gradienten sind für die Richtungsfindung nicht mehr unbedingt nötig. Es entsteht aber ein höherer Rechenaufwand, verglichen mit lokalen Optimierungen. [95, S. 60 ff.]

Beispielhaft sollen genetische Algorithmen als Teil der evolutionären Algorithmen nach [95, S. 62 f.] kurz erläutert werden. Ziel eines genetischen Algorithmus ist die globale Maximierung einer Zielfunktion. Basierend auf dem Prozess der Evolution werden verschiedene Punkte als sogenannte Individuen einer Bevölkerung definiert. Jedes Individuum besteht dabei aus verschiedenen Werten für die Variablen der Optimierung, welche häufig in mehrstellige binäre Variablen entsprechend natürlicher Chromosomen transformiert werden. Nun werden die Verfahren der iterativen Selektion und Manipulation auf eine Bevölkerung, also auf verschiedene Startpunkte angewandt. Bei der Selektion werden

stochastisch Individuen aus der Bevölkerung ausgewählt. Die Wahrscheinlichkeit der Auswahl eines Individuums wird dabei mittels dessen Maß der Anpassung definiert, welches dem Verhältnis des Zielfunktionswerts des Individuums und der summierten Zielfunktionswerte der Bevölkerung entspricht. Die genetische Manipulation eines Individuums erfolgt dann in zwei Schritten. Bei der Rekombination werden Chromosomen bei zwei zufällig ausgewählten Individuen getauscht. Bei der Mutation werden die Chromosomen eines Individuums selbst mit geringer Wahrscheinlichkeit verändert. Dieses Vorgehen wird solange wiederholt, bis das globale Optimum oder ein nach vorgegebenen Kriterien ausreichendes lokales Optimum gefunden wird. [95, S. 62 f.]

Für genetische oder evolutionäre Algorithmen gibt es wiederum viele unterschiedliche Implementierungen. In der vorliegenden Arbeit werden Algorithmen in der Programmiersprache *Python* für die globale Optimierung der hier behandelten Problemstellungen getestet. In *Python* kommt zum Beispiel wieder die Bibliothek *scipy* [124] in Frage. Zur Auswahl stehen beispielsweise die Algorithmen *basinhopping*, *brute force*, *dual annealing*, *differential evolution* und *simplicial homology global optimization (shgo)*. *Differential evolution* zählt zu den evolutionären Algorithmen und eignet sich für nicht-lineare, nicht-glatte und stetige Optimierungsprobleme. In [118] wird das Verfahren mit *simulated annealing* und genetischen Algorithmen verglichen und gezeigt, dass mit *differential evolution* im Vergleich häufig weniger Funktionsaufrufe notwendig sind. Eine relativ aktuelle Entwicklung ist der Algorithmus *shgo*, welcher in [35] mit anderen Ansätzen wie *Particle Swarm Optimization*, *Topographical Global Optimization*, *basinhopping* und *differential evolution* verglichen wird und dort gute Eigenschaften und kürzere Optimierungszeiten demonstriert. Der Algorithmus wurde für die Optimierung von *Lipschitz*-stetigen Zielfunktionen entwickelt, also stetige Funktionen mit beschränkter Steigung. Allerdings werden auch Lösungen für nicht-konvexe, nicht-glatte und auch nicht-stetige Funktionen gefunden [124].

2.3. Modellprädiktive Regelung

Eine Modellprädiktive Regelung, auch *Model Predictive Control* (MPC), optimiert die Steuergrößen eines Systems über einen zeitlichen Horizont unter Verwendung eines dynamischen Prozessmodells und wendet diese auf das System an. MPCs eignen sich für Mehrgrößensysteme, bei welchen sich mehrere manipulierbare Größen gegenseitig beeinflussen [32, S. 34]. Dies ist zum Beispiel bei Fernwärmenetzen der Fall. Die in dieser Arbeit zu optimierenden Variablen sind die Vorlauftemperatur und die Einspeiseleistung verschiedener Wärmezeugungsanlagen. Die Leistung wird dabei unter anderem durch die Vorlauftemperatur selbst und den Massenstrom beeinflusst. Eine Veränderung der Vorlauftemperatur hat somit direkt eine Auswirkung entweder auf die Einspeiseleistung oder auf den Massenstrom, welcher wiederum die gesamte Druckregelung ändert. Zudem sind MPCs auch die einzigen Algorithmen, „in denen Begrenzungen (constraints) für die Steuer- und Regelgrößen vorgegeben werden können, die im Regelalgorithmus selbst explizit und systematisch berücksichtigt werden“ [32, S. 34]. Durch die Verwendung eines Prozessmodells kann eine MPC den Verlauf der Systemzustände in der Zukunft vorhersagen und diesen optimieren. Auf zukünftige Abweichungen oder problematische Zustände kann bereits frühzeitig reagiert werden. Zudem kann kontinuierlich der optimale Betriebspunkt einer Anlage ermittelt und eingestellt werden. Auch diese Vorteile sind bei der optimalen Regelung verteilter Wärmeeinspeisung in Fernwärmenetzen relevant.

Im Folgenden soll das Funktionsprinzip einer MPC nach [32, S. 38 ff.] erläutert werden. Das Prinzip ist in Abbildung 2.8 vereinfacht illustriert. Die MPC soll zum Zeitpunkt k durchgeführt werden, an welchem der Sollwert w auf die dargestellte Größe springt. Hier wird ein konstanter Sollwert verwendet, dieser kann aber auch mit der Zeit geändert werden. Steuer- und Regel- sowie Messgrößen werden von der Regelung beobachtet und deren Verlauf gespeichert. Der MPC stehen zum Zeitpunkt k also die Historie der Regelgröße y und der Steuergröße u sowie potenziell Messwerte des aktuellen Systemzustandes und der Störgrößen zur Verfügung.

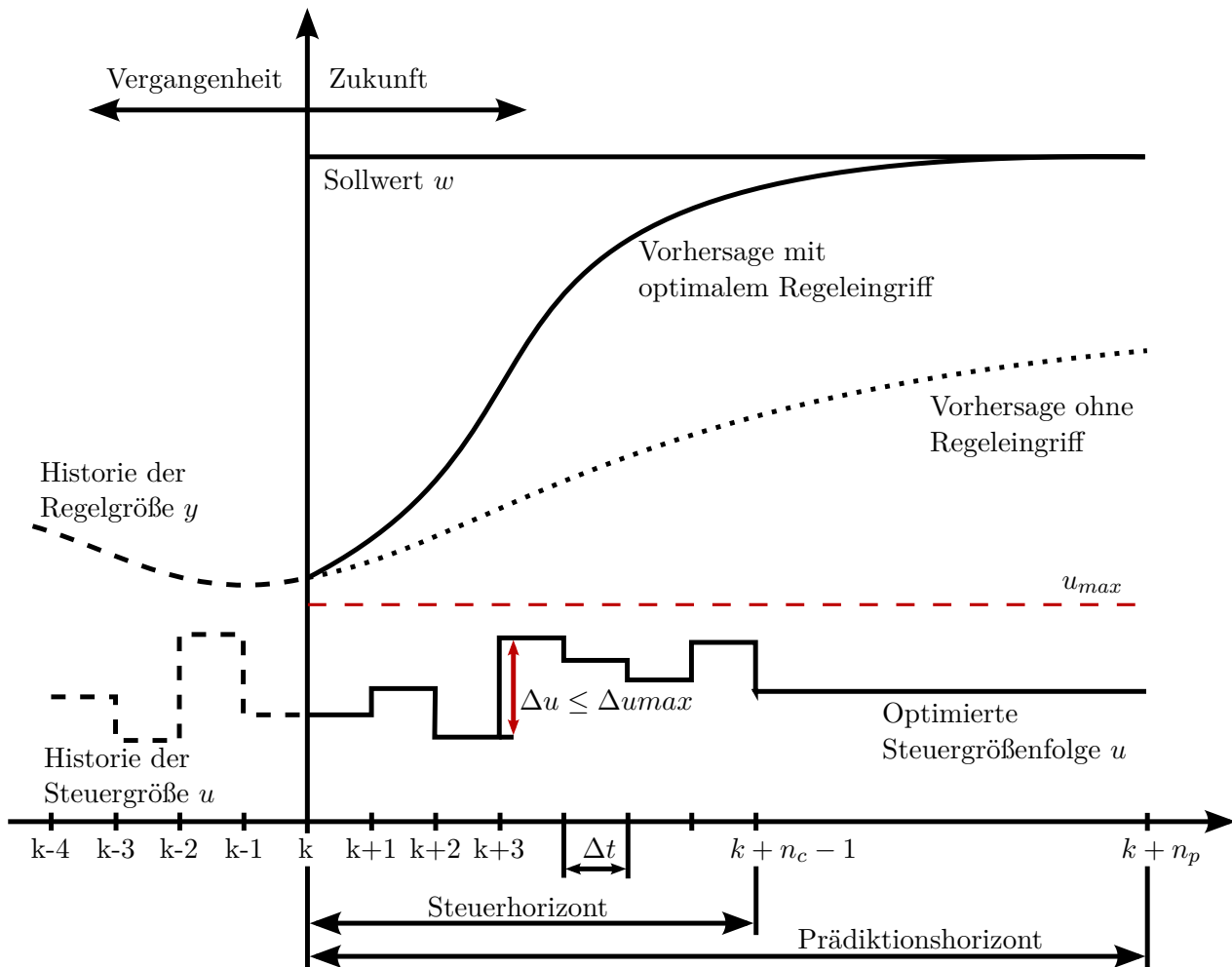


Abbildung 2.8.: Prinzip der Modellprädiktiven Regelung nach [32, S. 39].

Die MPC benötigt des Weiteren ein vollständiges Modell der Prozessdynamik, welches die Zusammenhänge zwischen allen Eingangs- und Ausgangsgrößen berechnet. Zunächst wird nun eine Simulation mit diskretem Zeitschritt Δt für den gesamten Prädiktionshorizont durchgeführt. Es werden also n_p Zeitschritte simuliert. Dabei wird die Steuergröße nicht verändert und im Zustand des Zeitpunktes k belassen. Ausgehend von diesem Systemverhalten können nun die Reglereingriffe für den Steuerhorizont n_c optimiert werden. Das Ziel der Optimierung ist dabei zum einen das Erreichen des Sollwertes w , zum anderen können weitere Restriktionen wie Grenzwerte der Zustands- oder Stellgrößen, vor allem auch maximale Gradienten der Stellgrößen, berücksichtigt werden. Das Systemverhalten kann ebenfalls bezüglich der Kosten modelliert und aus betriebswirtschaftlicher Sicht optimiert werden. [32, S. 39]

Nach der Optimierung der Stellgrößen können diese auf das System angewandt werden. Dabei gilt das „Prinzip des gleitenden Horizonts“ [32, S. 39]. Nur die erste der optimierten Stellgrößen wird aufgeschaltet. Anschließend wird der Steuer- und Prädiktionshorizont um einen Schritt verschoben und die Optimierung im nächsten Zeitschritt neu ausgeführt. Somit können aktuelle Prozessinformationen in jedem Schritt in der Optimierung berücksichtigt werden und der Regelkreis ist geschlossen. Die Optimierung für einen bestimmten Steuerhorizont und die Modellierung eines Prädiktionshorizonts ermöglicht ein frühzeitiges Feststellen zukünftiger Abweichungen des Systemverhaltens und deren Korrektur. Für das Erreichen bestimmter Sollwerte können somit optimale Trajektorien ermittelt und abgefahren werden.

2.4. Systemidentifikation

Die Systemidentifikation ist eine Technik für die Erstellung von Modellen komplexer Systeme basierend auf Messdaten. Dieser Ansatz wird zum Beispiel in der Regelungstechnik eingesetzt, um Modelle ohne den Aufwand komplexer physikalischer Beschreibungen der Systeme mit vergleichsweise geringen Rechenzeiten zu bilden. Auch in der Modellerstellung für MPCs wird Systemidentifikation häufig verwendet [32, S. 36].

Die Durchführung der Systemidentifikation kann in vier Schritte unterteilt werden [101, S. 17]:

1. Datensammlung
2. Auswahl der Modellstruktur
3. Berechnung der Modellparameter
4. Validierung des Modells

Bei der Datensammlung werden relevante Messdaten aufgenommen. Die Eingangssignale des Systems werden variiert und die benötigten Ausgangsdaten aufgenommen. Die Daten können hier zum Beispiel auch durch Simulationen mit komplexen physikalischen Modellen generiert werden.

Anschließend muss eine Modellstruktur gewählt werden. Dabei kommen allgemein *White-Box*, *Grey-Box* oder *Black-Box*-Ansätze in Frage [101, S. 17 f.]. Ein *White-Box*-Modell ist ein Modell, bei dem sämtliche physikalischen Zusammenhänge und Parameter bekannt sind. Für die Erstellung von *White-Box*-Modellen ist entsprechend eine sehr gute Kenntnis des Systems vorausgesetzt und Systemidentifikation nicht notwendig. Viel mehr sollen hier eben solche *White-Box*-Modelle durch Systemidentifikation ersetzt werden. Bei einem *Grey-Box*-Modell sind Teile der Zusammenhänge und Parameter bekannt. Die Unbekannten können durch Systemidentifikation abgeschätzt werden. *Black-Box*-Modelle versuchen hingegen, das Systemverhalten durch mathematische Gleichungen zu approximieren. Hier können beliebig einfache oder komplexe Funktionen gewählt und deren Parameter optimiert werden, um das reale Systemverhalten möglichst genau abzubilden. Sowohl lineare als auch nicht-lineare Gleichungen können bei der Systemidentifikation gewählt werden. [101, S. 17 f.]

Im dritten Schritt werden die Parameter der *Grey*- oder *Black-Box*-Modelle optimiert. Dafür ist eine Kostenfunktion notwendig, welche normalerweise den Unterschied zwischen den vom Modell vorhergesagten und den tatsächlich gemessenen Ausgangsdaten verwendet [101, S. 19]. Im letzten Schritt muss das erstellte Modell getestet und validiert werden.

Hier soll der zweite Schritt, die Auswahl der Modellstruktur, etwas genauer betrachtet werden. Anschließend werden auch Informationen zur Auswahl bzw. Identifikation der Modellparameter gegeben. Eine bekannte Implementierung der Algorithmen der Systemidentifikation ist die Systemidentifikationstoolbox in *MATLAB*[®]. Die folgenden Informationen über verschiedene Arten der Modellidentifikation wurden der *MATLAB*[®] Dokumentation in [123] und der Arbeit von Ljung [78], der maßgeblich an der Entwicklung der Systemidentifikationstoolbox beteiligt war, entnommen. Hier werden relevante Modelle, welche im späteren Teil der Arbeit für die Abbildung von Fernwärmenetzen genutzt werden, kurz erläutert.

2.4.1. Auswahl der Modellstruktur

Bei der Auswahl der Modellstruktur kommen Optionen der linearen und nicht-linearen Modelle infrage. In *MATLAB*[®] sind unter anderem Zustandsraummodelle, Übertragungsfunktionen und Polynommodelle verfügbar. Die Theorie dieser Modelle wird hier vorgestellt. Dabei handelt es sich zunächst um lineare, zeitinvariante Modelle (*Linear Time Invariant* (LTI)) [78, S. 18]. Die Systemantwort ist also

nicht von der Startzeit abhängig, das System muss linear sein und Parameter des Systems verändern sich nicht über die Zeit. Für Fernwärmenetze kann dies angenommen werden, da sich die physikalischen Parameter des Systems bis auf Verschleißerscheinungen nicht wesentlich ändern. Wie in Kapitel 5 später beschrieben wird, sind die thermischen und hydraulischen Zustandsgleichungen allerdings nicht-linear. Trotzdem sollen in dieser Arbeit zunächst auch lineare Modelle getestet werden, da sie einfacher zu trainieren sind und geringere Rechenzeiten aufweisen. So könnte ein Netz zum Beispiel auch an mehreren Arbeitspunkten linearisiert und somit durch eine Reihe linearer Modelle abgebildet werden. Im Anschluss werden auch nicht-lineare Ansätze, wie zum Beispiel nicht-lineare *AutoRegressive with exogenous terms* (ARX) Modelle getestet. Alle der untersuchten Modelle gehören zu den *Black-Box*-Ansätzen.

Zustandsraummodelle

Zustandsraummodelle beschreiben ein System mittels Differentialgleichungen erster Ordnung. Die Zustandsvariablen $\mathbf{x}(t)$ eines Systems können dabei aus gemessenen Eingangsgrößen $\mathbf{u}(t)$ und Ausgangsgrößen $\mathbf{y}(t)$ berechnet werden, zählen aber selbst nicht zu den Messwerten. Sind der Anfangszustand eines Systems und zukünftige Eingangsgrößen bekannt, können auch zukünftige Zustände und Ausgangsgrößen des Systems vollständig bestimmt werden. Zustandsgrößen können physikalisch als Speichergrößen wie Geschwindigkeiten, Temperaturen oder Ströme interpretiert werden [82, S. 75]. In *MATLAB*[®] muss für die Erstellung eines Zustandsraummodells neben den gemessenen Ein- und Ausgangswerten nur ein Parameter gewählt werden: die Modellordnung n . Diese entspricht der Dimension des Zustandsvektors $\mathbf{x}(t)$. Die Zustandsraumdarstellung lautet in *MATLAB*[®] folgendermaßen [123]:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{K}\mathbf{d}(t) \quad (2.41)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) + \mathbf{d}(t) \quad (2.42)$$

\mathbf{A} ist dabei die Systemmatrix, \mathbf{B} die Eingangsmatrix, \mathbf{C} die Ausgangsmatrix, \mathbf{D} die Durchgangsmatrix und \mathbf{K} die Störeingangsmatrix. \mathbf{d} beschreibt einen Störgrößenvektor. Die Matrizen enthalten jeweils Parameter, welche unbekanntem physikalischen Größen, Materialkonstanten etc. entsprechen [78, S. 93].

Da Messdaten nur zu bestimmten Zeitpunkten verfügbar sind, wird hier mit der zeitdiskreten Darstellung eines Zustandsraummodells gearbeitet. Dabei steht T für die Abtastzeit und kT für den aktuellen Zeitpunkt [123], vgl. [78, S. 98]:

$$\mathbf{x}(kT + T) = \mathbf{A}\mathbf{x}(kT) + \mathbf{B}\mathbf{u}(kT) + \mathbf{K}\mathbf{d}(kT) \quad (2.43)$$

$$\mathbf{y}(kT) = \mathbf{C}\mathbf{x}(kT) + \mathbf{D}\mathbf{u}(kT) + \mathbf{d}(kT) \quad (2.44)$$

Übertragungsfunktionen

Eine Übertragungsfunktion, auch *transfer function*, beschreibt den Zusammenhang zwischen Systemeingängen und -ausgängen im Frequenzbereich. Die allgemeine Darstellung der Übertragungsfunktion $G(s)$ mit der Laplace-Transformierten des Eingangs $u(s)$ und Ausgangs $y(s)$ lautet [78, S. 26]:

$$G(s) = \frac{y(s)}{u(s)} \quad (2.45)$$

Die Übertragungsfunktion kann mithilfe von Polynomen im Zähler ($Z(s)$) und Nenner ($N(s)$) beschrieben und durch eine Störgröße $d(s)$ sowie eine Verzögerung τ zwischen Ein- und Ausgang ergänzt

werden. Ein Übertragungsfunktionsmodell in *MATLAB*[®] hat die folgende Form [123]:

$$y(s) = \frac{Z(s)}{N(s)} e^{-s\tau} u(s) + d(s) \quad (2.46)$$

Dabei wird die Modellordnung mit der Ordnung der Polynome im Nenner bestimmt, die Nullstellen des Nenners bestimmen die Pole, die Nullstellen des Zählers die Nullstellen des Systems. Als Parameter eines Modells können die Anzahl der Pole und Nullstellen sowie die Verzögerung gewählt werden.

Polynommodelle

Anwendung in der Systemidentifikation finden ebenfalls Polynommodelle, welche mit den Eingängen \mathbf{u} , dem Ausgang y und der Störgröße d in *MATLAB*[®] folgendermaßen formuliert werden [123], vgl. [78, S. 87]:

$$A(q^{-1})y(t) = \sum_{i=1}^{\nu} \frac{B_i(q^{-1})}{F_i(q^{-1})} u_i(t - nk_i) + \frac{C(q^{-1})}{D(q^{-1})} d(t) \quad (2.47)$$

ν ist dabei die Anzahl der Eingänge, nk_i die Verzögerung des jeweiligen Eingangs. A , B , C , D und F sind Polynome beliebiger Form, welche in Abhängigkeit des *time-shift*-Operators q^{-1} angegeben werden, zum Beispiel [123], vgl. [78, S. 81]:

$$B(q^{-1}) = b_0 + b_1 q^{-1} + \dots + b_{nb} q^{-nb} \quad (2.48)$$

$$\text{mit } q^{-1} u_i(t) = u_i(t - T) \quad (2.49)$$

$$q^{-2} u_i(t) = u_i(t - 2T) \quad (2.50)$$

⋮

$$q^{-nb} u_i(t) = u_i(t - nbT) \quad (2.51)$$

Je nach Polynommodell können einige der Polynome vernachlässigt und die Darstellung und Handhabung vereinfacht werden. Andererseits ermöglicht die Verwendung der zum Beispiel im Vergleich zur Übertragungsfunktion zusätzlichen Polynome eine größere Flexibilität. Ein ARX Modell verwendet nur die Polynome A und B , ein *AutoRegressive-Moving Average with eXogenous terms* (ARMAX) Modell zusätzlich C und ermöglicht eine bessere Darstellung der Störgröße. *Box-Jenkins*-Modelle enthalten alle Polynome außer A . Die Modelle werden ausführlich in dem Buch von Ljung [78, S. 81 ff.] erläutert. Zusätzlich kann ein Integrator in der Störgröße verwendet werden und verändert die allgemeine Darstellung [123]:

$$A(q^{-1})y(t) = \sum_{i=1}^{\nu} \frac{B_i(q^{-1})}{F_i(q^{-1})} u_i(t - nk_i) + \frac{C(q^{-1})}{D(q^{-1})} \frac{1}{1 - q^{-1}} d(t) \quad (2.52)$$

Ein *AutoRegressive Integrated with eXogenous terms* (ARIX) Modell ist entsprechend ein ARX Modell erweitert um den Integrator und ein *AutoRegressive Integrated Moving Average with eXogenous terms* (ARIMAX) Modell das gleiche für ein ARMAX Modell.

Nicht-lineare ARX Modelle

Basierend auf Polynommodellen können ebenfalls nicht-lineare Modelle erstellt werden. Die beschriebene Darstellung für ein lineares ARX Modell mit einem um n_k verzögerten Eingang u entspricht der Differenzgleichung eines Systems mit einem Ein- und einem Ausgang [123] [78, S. 8]:

$$y(t) + a_1y(t-1) + a_2y(t-2) + \dots + a_{na}y(t-na) = b_1u(t-n_k) + b_2u(t-n_k-1) + \dots + b_{nb}u(t-n_k-nb+1) + e(t) \quad (2.53)$$

Der derzeitige Ausgangswert $y(t)$ wird basierend auf dem derzeitigen Eingangswert $u(t)$ und den na vorangegangenen Eingangs- sowie nb vorherigen Ausgangswerten bestimmt. Die Gleichung kann folgendermaßen auch als Produkt formuliert werden [123]:

$$y_p(t) = [-a_1, -a_2, \dots, -a_{na}, b_1, b_2, \dots, b_{nb}] \cdot [y(t-1), y(t-2), \dots, y(t-na), u(1-n_k), u(t-n_k-1), \dots, u(t-n_k-nb+1)]^T \quad (2.54)$$

$y(t-1)$, $u(t-n_k)$ etc. sind dabei die verzögerten Ein- und Ausgänge, welche auch Regressoren genannt werden. Die Koeffizienten wie a_1 oder b_1 beschreiben die Gewichtung des jeweiligen Regressors. Ein lineares ARX Modell ermittelt somit dessen Ausgangsgröße als gewichtete Summe der Regressoren. Bei nicht-linearen ARX Modellen können komplexere nicht-lineare Funktionen für die Regressoren gewählt werden. Diese *Mapping*-Funktionen F haben als Eingang die Regressoren [123]:

$$y_p(t) = F(y(t-1), y(t-2), \dots, y(t-na), u(1-n_k), u(t-n_k-1), \dots, u(t-n_k-nb-1)) \quad (2.55)$$

Auch für die Regressoren selbst können nicht-lineare Beziehungen gewählt werden. Insgesamt besteht ein nicht-lineares ARX Modell aus den Regressoren und einer Ausgangsfunktion, welche wiederum aus linearen und nicht-linearen Funktionen sowie einem Versatz (*Offset*) zusammengesetzt wird. Die Struktur eines solchen Modells ist in Abbildung 2.9 dargestellt.

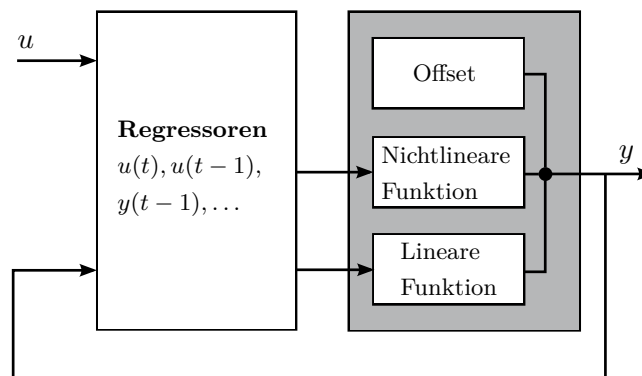


Abbildung 2.9.: Struktur eines nicht-linearen ARX Modells nach [123].

Weitere nicht-lineare Modelle

Weitere nicht-lineare Modelle sind zum Beispiel *Hammerstein-Wiener*-Modelle, welche verwendet werden können, wenn eine Zerlegung der Nichtlinearität möglich ist. Verschiedene Blöcke werden dabei seriell verbunden. Zentral ist ein lineares Modell, welchem eine Nichtlinearität als Eingang vorge-schaltet und / oder als Ausgang nachgeschaltet wird. Ein solches Modell kann als *Black-* oder *Grey-Box*-Modell verwendet werden. Des Weiteren können in *MATLAB*[®] nicht-lineare *Grey-Box*-Modelle erstellt werden, indem das System durch Differential- oder Differenzgleichungen beschrieben wird und anschließend deren Parameter identifiziert werden.

2.4.2. Berechnung der Modellparameter

Nachdem ein Modell und dessen Struktur festgelegt wurden, folgt der nächste Schritt der Systemidentifikation: die Berechnung der Modellparameter. Die Parameter eines Modells müssen identifiziert werden. Hier kommen verschiedene Methoden wie zum Beispiel der *prediction-error*-, der *correlation*- oder der *subspace*-Ansatz in Frage, welche in [78, S. 197 ff.] vorgestellt werden. Im Prinzip werden Modelle anhand ihrer Vorhersagewerte, verglichen zu den in Schritt 1 gesammelten Daten, bewertet. Der Vorhersagefehler soll dabei durch die Auswahl geeigneter Parameter minimiert werden. Der *prediction-error-identification*-Ansatz wird in diesem Abschnitt nach Ljung als ein Beispiel für die Funktionsweise der Systemidentifikation erläutert. Die Menge an möglichen Modellen \mathcal{M}^* mit dem Parametervektor $\boldsymbol{\theta} \in \mathcal{D}_{\mathcal{M}} \subset \mathbb{R}^d$ ist [78, S. 197]:

$$\mathcal{M}^* = \{\mathcal{M}(\boldsymbol{\theta}) | \boldsymbol{\theta} \in \mathcal{D}_{\mathcal{M}}\} \quad (2.56)$$

$\mathcal{D}_{\mathcal{M}}$ steht für die Menge der möglichen Parametervektoren eines Modells \mathcal{M} . Der Vorhersagefehler $\epsilon(t, \boldsymbol{\theta})$ wird jeweils mit einem gesammelten Ausgangswert y und einem vom Modell vorhergesagter Ausgangswert \hat{y} berechnet [78, S. 198].

$$\epsilon(t, \boldsymbol{\theta}) = y(t) - \hat{y}(t, \boldsymbol{\theta}) \quad (2.57)$$

Der Vorhersagefehler wird anschließend mit einem linearen Filter $L(q)$ gefiltert [78, S. 199]:

$$\epsilon_F(t, \boldsymbol{\theta}) = L(q)\epsilon(t, \boldsymbol{\theta}) \quad (2.58)$$

Die folgende Norm V_N mit der skalarwertigen Funktion ℓ wird auf den gefilterten Fehler angewandt [78, S. 199]:

$$V_N(\boldsymbol{\theta}, \mathbf{Z}^N) = \frac{1}{N} \sum_{t=1}^N \ell(\epsilon_F(t, \boldsymbol{\theta})) \quad (2.59)$$

\mathbf{Z}^N ist dabei der gesammelte Satz an N Ein- und Ausgangsdaten [78, S. 198]:

$$\mathbf{Z}^N = [u(1), \dots, u(N), y(1), \dots, y(N)] \quad (2.60)$$

Anschließend wird die skalarwertige Funktion V_N , welche von den Modellparametern $\boldsymbol{\theta}$ und dem Datensatz \mathbf{Z}^N abhängt, minimiert, um den Parametersatz $\hat{\boldsymbol{\theta}}_N$ zu erhalten [78, S. 199]:

$$\hat{\boldsymbol{\theta}}_N = \hat{\boldsymbol{\theta}}_N(\mathbf{Z}^N) = \arg \min_{\boldsymbol{\theta} \in \mathcal{D}_{\mathcal{M}}} V_N(\boldsymbol{\theta}, \mathbf{Z}^N) \quad (2.61)$$

Das numerische Lösen des Minimierungsproblems der *prediction-error-identification*-Methode und anderer Methoden kann durch verschiedene Verfahren wie zum Beispiel die lineare Regression, die Methode der kleinsten Quadrate oder das Gradientenverfahren erfolgen. Die Verfahren werden in [78, S. 317 ff.] und zum Teil auch in Kapitel 2.2 beschrieben.

Insgesamt wird ersichtlich, dass für die Systemidentifikation sowohl eine Vielzahl an Modellen als auch Methoden für deren Identifikation vorhanden sind. Die Grundlagen wurden zunächst prinzipiell erläutert und werden später in der Arbeit in Kapitel 6 für die Erstellung von Fernwärmenetzmodellen verwendet.

2.5. Maschinelles Lernen

Methoden des maschinellen Lernens werden in dieser Arbeit als Alternative zur Systemidentifikation verwendet, um Modelle komplexer Systeme zu vereinfachen. Das maschinelle Lernen kann als ein Teilgebiet der künstlichen Intelligenz bezeichnet werden, bei dem Algorithmen verwendet werden, um aus Daten zu lernen. So kann zum Beispiel ein Algorithmus basierend auf vielen Beispielen lernen, Gesichter in Bildern zu erkennen. In Bezug auf die vorliegende Arbeit soll ein Algorithmus basierend auf Beispieldaten von Fernwärmenetzen lernen, relevante hydraulische und thermische Zustandsgrößen abhängig von den Regelgrößen vorherzusagen. So soll ein komplexes physikalisches Fernwärmenetzmodell durch ein automatisch gelerntes (abstraktes) Modell ersetzt werden, welches durch geringe Rechenzeiten für eine Optimierung geeignet ist. Die Bedienung der Modelle wäre die gleiche: Dem erlernten Modell werden die gleichen oder sogar reduzierte Eingangsdaten wie dem physikalischen Modell übergeben, um damit die relevanten Ausgangsgrößen für eine Regelung oder Optimierung zu berechnen.

Verschiedene Methoden kommen infrage, welche unterschiedliche Modellstrukturen beschreiben und deren Parameter im Lernprozess mittels gemessener Ein- und Ausgangsdaten optimiert werden. Im Prinzip funktioniert das maschinelle Lernen also ähnlich der Systemidentifikation. Die Systemidentifikation basiert dabei allerdings auf anderen Lernansätzen und ist hauptsächlich auf das Lernen dynamischer Systeme ausgelegt, während bei maschinellem Lernen unterschiedliche Anwendungsgebiete existieren. Zudem ist bei der Systemidentifikation die Struktur des Modells bereits vor dem Lernen festgelegt. Bei maschinellem Lernen bleibt dem Lernalgorithmus und Anwender in dieser Hinsicht eine größere Flexibilität. Im Weiteren wird eine Einführung und Zusammenfassung der unterschiedlichen Ansätze gegeben und eine prinzipielle Überprüfung ihrer Anwendbarkeit auf das vorliegende Problem durchgeführt.

Maschinelles Lernen kann zunächst in strukturiertes und unstrukturiertes Lernen unterteilt werden. Strukturierte Daten sind zum Beispiel Daten einer Tabelle, in welcher ein klarer Zusammenhang zwischen Merkmalen (Eingangsgrößen) und Ausgangsgrößen besteht. Als unstrukturierte Daten können beispielsweise Bilder bezeichnet werden, deren Inhalt nicht in Form von Text beschrieben ist. Hier werden nur strukturierte Daten verwendet. Allgemein sind viele Datensätze, in welchen das gesamte Systemverhalten abgedeckt wird, für das Lernen von Vorteil. Weist ein Problem viele Merkmale, hier also viele Variablen auf, muss auf den Fluch der Dimensionalität (*Curse of Dimensionality*) nach Bellman hingewiesen werden [36, S. 333 f.] [42, S. 20]. Mit diesem Begriff werden Probleme bezeichnet, welche bei Datensätzen mit vielen Dimensionen bzw. in diesem Fall vielen Optimierungsvariablen entstehen und zu hohen Rechenzeiten führen. Dieses Problem tritt zum Beispiel beim maschinellen Lernen auf und ist für die Optimierung besonders von Fernwärmenetzen mit vielen dezentralen Wärmeinspeisungen relevant.

Maschinelles Lernen kann weiterhin in überwachtes, unüberwachtes und bestärkendes Lernen eingeteilt werden [42, S. 21]. Bei allen Lernalgorithmen ist das Ziel, eine mathematische Abbildung zu erlernen [42, S. 21]:

$$f : \mathbf{X} \rightarrow \mathbf{Y} \quad (2.62)$$

Beim überwachten Lernen werden dem Algorithmus komplette Datensätze, bei denen sowohl die Eingangsdaten \mathbf{X} als auch die Ausgangsdaten \mathbf{Y} bekannt sind, zur Verfügung gestellt. Beim unüberwachten Lernen sind die Ausgangsdaten \mathbf{Y} nicht bekannt. Es wird lediglich ein Datensatz von Merkmalen in einen Algorithmus gegeben, welcher dann versucht, Strukturen in diesen Daten zu finden [42, S. 25]. Beim bestärkenden Lernen (*Reinforcement-Learning*) ist der optimale Ausgang eines Algorithmus nicht bekannt, allerdings kann das Ergebnis als positiv oder negativ bewertet werden. Basierend darauf wird eine Strategie erlernt [42, S. 24]. In der vorliegenden Arbeit soll das thermische

und hydraulische Verhalten von Fernwärmenetzen erlernt werden. Dies kann entweder basierend auf Mess- oder Simulationsdaten erfolgen. In beiden Fällen sind die Eingangsgrößen \mathbf{X} ebenso wie die relevanten Ausgangsdaten \mathbf{Y} bekannt. Hier handelt es sich also um überwachtes Lernen von strukturierten Daten.

Das überwachte Lernen umfasst zwei unterschiedliche Problemstellungen: Regression bzw. Approximation und Klassifikation [36, S. 193 f.]. Bei der Klassifikation geht es darum, Daten zu klassifizieren, und die Zieldaten bzw. die Zielmenge \mathbf{Y} ist diskret. Bei der Regression ist die Menge $\mathbf{Y} \subseteq \mathbb{R}^n$ und die Werte sind somit kontinuierlich [42, S. 23] [36, S. 194]. Ein Regressionsproblem ist zum Beispiel, den Schlechtpunktdruck in einem Fernwärmenetz, basierend auf Eingangsdaten wie dem Vorlaufdruck, der Vorlauftemperatur und dem Massenstrom zu bestimmen. In dieser Arbeit handelt es sich also um Probleme der Regression. Allgemein basieren die Algorithmen des maschinellen Lernens wieder auf einer Optimierung bzw. Minimierung. Als Zielfunktion der Optimierung wird dabei die Abweichung der von einem Algorithmus vorhergesagten und der tatsächlich gemessenen Ausgangsdaten verwendet.

Vorbereitung der Daten

Bevor nun Daten für das Training von Modellen verwendet werden, sollten diese zunächst vorbereitet werden. Dazu gehört zum Beispiel das Löschen von Datenanomalien. Auch kann der Einfluss der Eingangs- auf die Ausgangsdaten a priori zum Beispiel durch eine Sensitivitätsanalyse untersucht werden [72, S. 78 f.], um den Datensatz eventuell um solche Merkmale zu reduzieren, die kaum einen Einfluss auf die Ausgänge haben. Da einige Lernverfahren empfindlich gegenüber besonders großen oder kleinen Eingangswerten sind und es zu numerischen Problemen durch unterschiedlich skalierte Daten kommen kann, können die Daten vor dem Lernen normiert oder standardisiert werden [72, S. 41 f.] [42, S. 275 ff.]. Bei der Normierung werden die Daten in einem gewünschten Eingangswertebereich skaliert, zum Beispiel in den Intervallen $[0, 1]$ oder $[-1, 1]$. Bei der Standardisierung werden die Eingangsdaten stattdessen so verteilt, dass das arithmetische Mittel der Daten 0 und deren Varianz 1 ist [42, S. 278]. Die Standardisierung ist zum Beispiel bei Daten mit Anomalien sinnvoll, da in diesem Fall bei einer Normierung auf das Intervall $[-1, 1]$ ein großer Teil der Daten nur noch in einem sehr kleinen Zahlenbereich liegen würde [42, S. 280]. Auch Ausgangsdaten können vor dem Lernen normiert oder standardisiert werden. Dies ist besonders bei unterschiedlich großen Ausgangsdaten sinnvoll, da sonst größere Ausgangswerte mit absolut größeren Fehlern stärker in der Zielfunktion einer Optimierung berücksichtigt würden.

Überanpassung

Ein wichtiger Punkt beim Maschinellen Lernen ist die Unterteilung der Daten in eine Trainings-, Validierungs- und Testmenge [42, S. 84 ff.] [72, S. 76 f.]. Lernalgorithmen neigen dazu, die Trainingsmenge bei vielen Freiheitsgraden sehr gut abzubilden. Angewandt auf neue, unbekannte Daten ist dies möglicherweise nicht mehr der Fall. Dieses Phänomen wird Überanpassung (*Overfitting*) genannt [72, S. 77]. Eine Unterteilung in eine Trainings- und Validierungsmenge sowie eine unabhängige und ausreichend große Testmenge verhindert eine Überanpassung, indem nur ein Teil der Daten für das Training und ein Teil der Daten für das Testen eines Algorithmus verwendet wird. Eine Validierungsmenge wird verwendet, falls die Validierung verschiedener Parameter oder Hyperparameter während des Trainings notwendig ist. Auch wenn eine Reihe von Modellen trainiert wird, können diese Daten verwendet werden, um daraus das beste Modell auszuwählen. Zudem besteht die Option, das Training frühzeitig abubrechen (*Early Stopping*) [72, S. 78] [42, S. 208 f.], wenn zum Beispiel die Genauigkeit bezüglich der Trainingsdaten noch zunimmt, die der Validierungsdaten aber abnimmt. Nach dem Training kann die Qualität eines Algorithmus mittels der unabhängigen Testmenge bestimmt werden. Eine typische Aufteilung der Daten ist 70 bis 85 % für die Trainingsmenge und 15 bis 30 % für die Testmenge [42, S. 85] oder wenn zusätzlich eine Validierungsmenge benötigt wird in 60 %, 20 % und 20 % für die Trainings-, Validierungs- und Testmenge [42, S. 209].

2.5.1. Algorithmen des maschinellen Lernens

Verschiedene Bibliotheken für das maschinelle Lernen sind in vielen Programmiersprachen vorhanden. Beliebt sind zum Beispiel *Python* und *R*. Auch *MATLAB*[®] verfügt über entsprechende Bibliotheken. In dieser Arbeit soll im Folgenden *Python* 3.8.12 verwendet werden. Hier kommen beispielsweise die quelloffene Bibliothek *Keras* für tiefe neuronale Netze, welche auf dem von Google entwickelten *TensorFlow* (hier in Version 2.3.0) basiert, oder die Bibliothek *PyTorch* (Version 1.11.0) infrage. Für die Erstellung von linearen Modellen oder Entscheidungsbäumen wird *scikit-learn* in der Version 1.0.2 verwendet. Auch *Boosting*-Algorithmen wie zum Beispiel *XGBoost* (Version 1.5.2) stehen in *Python* zur Verfügung. Im Folgenden werden einige der gängigen Algorithmen angefangen mit einfachen statistischen Methoden bis zu komplexeren Algorithmen sowie deren wichtigste Eigenschaften und Parameter vorgestellt. Anschließend werden diese Algorithmen in Kapitel 5 bezüglich ihrer Anwendung auf Fernwärmenetze getestet.

2.5.2. Lineare Modelle

Eine einfache, aus der Statistik bekannte Methode ist die lineare Regression. Bei der linearen Regression wird von einem linearen Zusammenhang zwischen den Eingangsdaten \mathbf{x} und den Ausgangsdaten \mathbf{y} ausgegangen. Dieser gesuchte Zusammenhang $y = f(\mathbf{x})$ kann bei einer Ausgangsvariablen y , mehreren Eingangsvariablen \mathbf{x} und den Koeffizienten \mathbf{u} durch folgende Funktion beschrieben werden [42, S. 115]:

$$f(\mathbf{x}) = u_0 + u_1 \cdot x_1 + \dots + u_n \cdot x_n = u_0 + \sum_{i=1}^n u_i \cdot x_i \quad (2.63)$$

Für die Koeffizienten werden bei der linearen Regression nun Werte gesucht, welche die Abweichung zwischen dem Vektor der berechneten Werte $\hat{\mathbf{y}}$ und dem Vektor der tatsächlichen Daten \mathbf{y} minimieren. Hier kann die Methode der kleinsten Fehlerquadrate angewandt werden, um den folgenden Term zu minimieren [42, S. 117]:

$$\sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2.64)$$

Eine weitere lineare Methode ist der *k-Nearest-Neighbor*-Algorithmus, welcher in [42, S. 122 ff.] und [36, S. 206 ff.] beschrieben wird. Der Algorithmus zählt zu den Verfahren des *Lazy-Learnings*, bei welchen die Hauptarbeit während der Anwendung geleistet wird und nicht, wie bei den *Eager-Learning*-Verfahren, während des Trainings. Dies bietet den Vorteil weniger einzustellender Parameter, allerdings wird auch bei der Anwendung der gesamte Trainingsdatensatz benötigt und die Vorhersage dauert länger [36, S. 214 f.]. Für das vorliegende Problem scheint dies kein geeigneter Ansatz zu sein. *Eager-Learning*-Verfahren sollten eingesetzt werden, da eine möglichst geringe Rechenzeit der Modelle für die anschließende Optimierung wichtig ist. Die lineare Regression kann hingegen getestet werden, um einfache lineare Modelle zu erstellen. Aufgrund der nicht-linearen Zusammenhänge in Fernwärmenetzen ist allerdings davon auszugehen, dass das Verhalten eines Netzes nicht mit einem linearen Modell allein korrekt abgebildet werden kann.

2.5.3. Entscheidungsbäume

Ein Entscheidungsbaum besteht aus einer Wurzel W , Knoten t und Blättern b [42, S. 129]. Ein binärer Entscheidungsbaum ist beispielhaft in Abbildung 2.10 dargestellt. Eine Vorhersage beginnt an der Wurzel. An dieser und anschließend an jedem Knoten werden Entscheidungen getroffen, bis der Algorithmus schließlich ein Blatt erreicht. Dieses Blatt beinhaltet das Ergebnis des Entscheidungsbaumes. Beim Trainieren eines solchen Baumes werden dessen Struktur sowie die Entscheidungskriterien an der Wurzel und an den Knoten festgelegt.

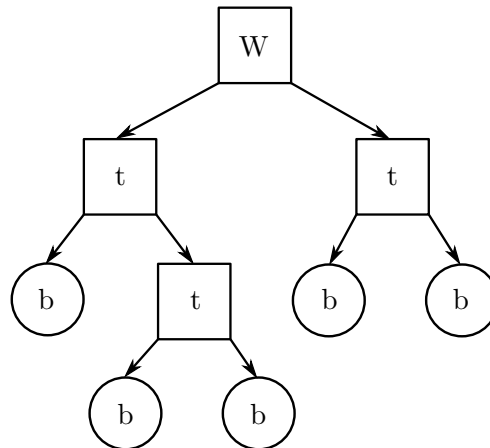


Abbildung 2.10.: Binärer Entscheidungsbaum nach [42, S. 129].

Auch Entscheidungsbäume haben das Problem der Überanpassung, wenn komplexe Bäume mit vielen Verzweigungen trainiert werden [36, S. 203]. Dabei ist ein großer Vorteil von Bäumen deren Transparenz. Es kann genau nachvollzogen werden, warum eine Entscheidung getroffen wird. Bei komplexen Bäumen ist dies nicht mehr so einfach möglich und sie führen bei verrauschten Daten zu größeren Fehlern als kleinere Bäume [42, S. 163]. Eine Lösung für dieses Problem ist das Stutzen bzw. *Pruning* der Bäume durch Entfernen einiger Verzweigungen. Beim *Pre-Pruning* wird vorher festgelegt, wann das Lernen abgebrochen werden soll, sodass keine zu komplexen Strukturen entstehen. Beim *Post-Pruning* wird nach abgeschlossenem Lernen gestutzt. [42, S. 162 ff.] [36, S. 229 ff.]

2.5.4. Neuronale Netze

Neuronale Netze basieren ursprünglich auf vereinfachten Modellen des menschlichen Gehirns [72, S. 7]. Sie bestehen aus einem Eingangs- bzw. Merkmalsvektor, Neuronen, welche in mehreren Schichten aufgeteilt werden können und einem Ausgangsvektor. Die Neuronen bestehen wiederum aus Gewichten und einer Aktivierungsfunktion. Abbildung 2.11 zeigt ein solches Neuron.

In einem Neuron wird die Funktion $h(\mathbf{x})$ (2.65) aus einem Eingangsvektor \mathbf{x} mit n Gewichten $\mathbf{w} \in \mathbb{R}$ gebildet. Die Funktion ist anschließend Eingang für die Aktivierungsfunktion g , welche einen Ausgangswert ab einem bestimmten Schwellwert von $h(\mathbf{x})$ entsprechend dem Schwellenpotenzial einer Nervenzelle weitergibt [42, S. 169] [36, S. 268 f.]:

$$h(\mathbf{x}) = \sum_{i=0}^n w_i \cdot x_i \quad (2.65)$$

Ein beispielhaftes neuronales Netz, hier spezifisch ein mehrschichtiges Perzeptron [72, S. 43 ff.], ist in Abbildung 2.12 dargestellt. Dieses besteht aus einem Eingangsvektor mit drei Werten, welche im *Input-Layer* über drei Neuronen in das neuronale Netz eingehen. Anschließend ist hier eine weitere versteckte

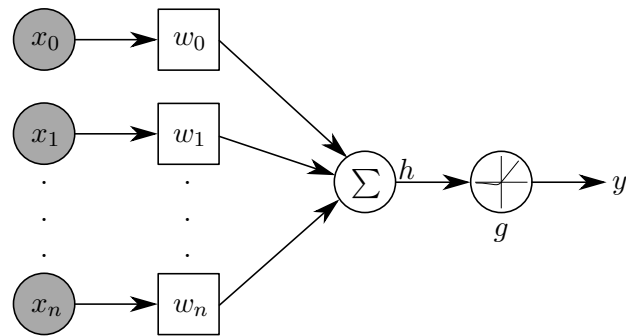
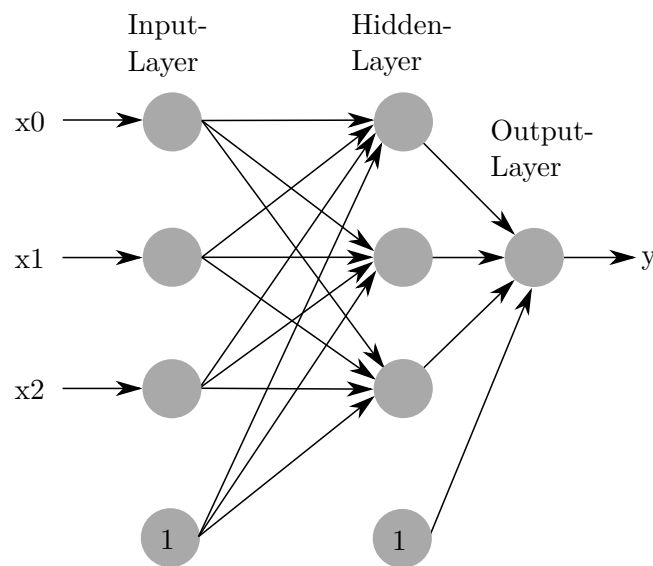


Abbildung 2.11.: Modell eines Neurons nach [42, S. 169].

Schicht, ein sogenannter *Hidden-Layer*, vorhanden, welcher wiederum aus drei Neuronen besteht. Dem *Input*- sowie jeder versteckten Schicht wird jeweils ein zusätzliches Neuron mit einem konstanten Wert, hier 1, hinzugefügt. Dieses sogenannte *Bias*-Neuron erlaubt es einer Schicht des neuronalen Netzes einen Versatz hinzuzufügen. Die Anzahl der versteckten Schichten und deren Neuronen kann beim Training eines neuronalen Netzes gewählt werden. Die Anzahl der Neuronen des *Input*- und *Output-Layers* ist durch die Anzahl der Ein- und Ausgänge festgelegt. Neuronale Netze mit mehr als zwei versteckten Schichten [42, S. 181] und / oder Anwendung von Ansätzen wie *Convolutional Neural Networks* [36, S. 300] werden tiefe neuronale Netze oder *deep neural networks* genannt, daher auch der Begriff *deep learning*.

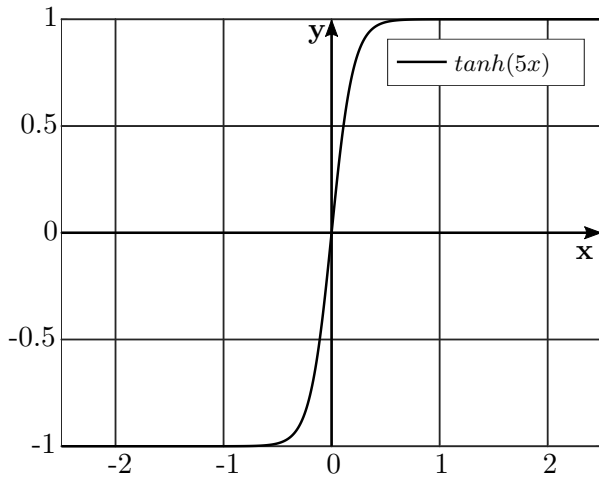
Abbildung 2.12.: Neuronales Netz mit einem *Hidden-Layer* nach [42, S. 179] [72, S. 44].

Aktivierungsfunktionen

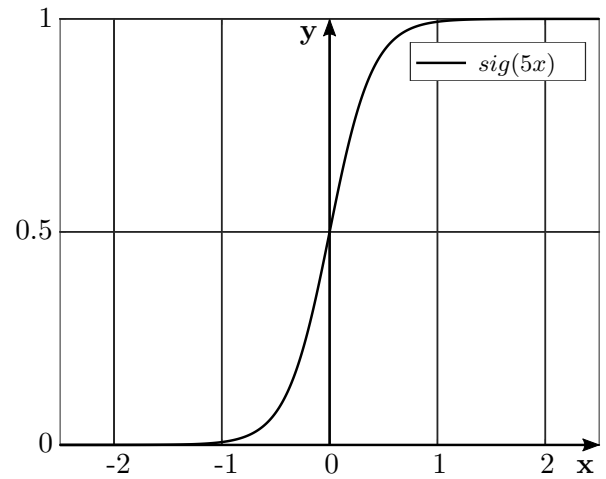
Neben der Anzahl der versteckten Schichten und Neuronen können unterschiedliche Aktivierungsfunktionen gewählt werden. Häufig verwendete Aktivierungsfunktionen sind der *Tangens Hyperbolicus* und die *Sigmoidfunktion*, dargestellt in den Abbildungen 2.13a und 2.13b. Der Ausgangswert φ der beiden Funktionen kann folgendermaßen berechnet werden [42, S. 178]:

$$\varphi = \tanh(a \cdot x) \quad (2.66)$$

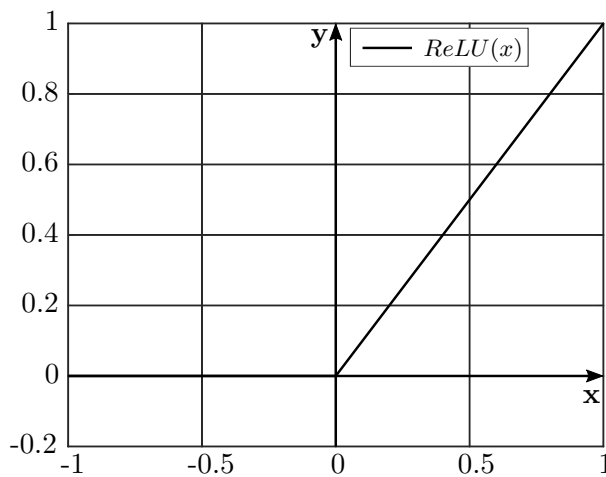
$$\varphi = \text{sig}(a \cdot x) = \frac{1}{1 + e^{-a \cdot x}} \quad (2.67)$$



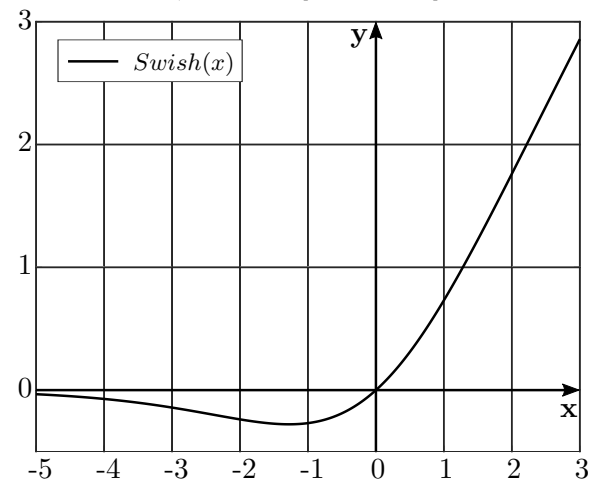
a) Tangens Hyperbolicus [42, S. 178]



b) Sigmoid [42, S. 178]



c) ReLU [42, S. 226]



d) Swish [103]

Abbildung 2.13.: Verschiedene Aktivierungsfunktionen für die Neuronen der neuronalen Netze.

Aufgrund der für das Training von neuronalen Netzen verwendeten Gradientenverfahren kann bei diesen Funktionen allerdings ein Problem auftreten. Bei großen Eingangswerten verschwinden die Gradienten und eine weitere Optimierung ist kaum noch möglich [42, S. 226]. Dies ist vor allem bei tiefen neuronalen Netzen ein Problem. Im Gegensatz dazu ist zum Beispiel der Gradient der Aktivierungsfunktion *Rectifier Linear Unit* (ReLU) stückweise konstant und weist nicht die Probleme verschwindender Gradienten auf. Daher ist ReLU derzeit die am häufigsten verwendete Aktivierungsfunktion für neuronale Netze [103, S. 1]. Einige Studien versuchen, die Funktion ReLU (Abbildung 2.13c) noch weiter zu verbessern, wie zum Beispiel durch die Funktion *Swish* (Abbildung 2.13d), welche mittlerweile ebenfalls häufig verwendet wird und im Gegensatz zu ReLU eine glatte Funktion ist. Die Funktionen ReLU und *Swish* sind folgendermaßen definiert [42, S. 226] [103]:

$$\text{ReLU}(x) = \begin{cases} x & \text{für } x \geq 0 \\ 0 & \text{für } x < 0 \end{cases} \quad (2.68)$$

$$\text{Swish}(x) = x \cdot \text{sig}(x) = x \cdot \frac{1}{1 + e^{-x}} \quad (2.69)$$

Zielfunktion und Optimierung

Für das Lernen kommen verschiedene Zielfunktionen (auch *loss*-Funktionen) und Optimierungsalgorithmen infrage. Beim Lernen werden die Gewichte und *Bias*-Werte so optimiert, dass eine Zielfunktion minimiert wird. Die anderen Parameter wie die Anzahl der versteckten Schichten und Neuronen sowie die Aktivierungsfunktion werden bereits vorher gewählt. Als Zielfunktion J der Gewichte \mathbf{W} wird häufig die Summe der quadratischen Fehler der Vorhersage mit der Trainingsmenge $\mathbf{D} = \{\mathbf{X}, \mathbf{Y}\}$ verwendet [42, S. 183]:

$$J(\mathbf{W}) = \sum_{i=1}^n \frac{1}{2} \cdot (y_i - f(\mathbf{x}_i))^2 \quad (2.70)$$

Auch andere Zielfunktionen wie der durchschnittliche absolute Fehler kommen infrage. Für die Optimierung bzw. die Bestimmung der Parameter eines neuronalen Netzes werden Gradientenverfahren verwendet [72, S. 58] [42, S. 183]. Deren Funktionsweise wurde in Kapitel 2.2 erläutert. Im Grunde folgen diese Verfahren, ausgehend von einem Startpunkt der Gewichte, immer wieder der steilsten Abstiegsrichtung, welche anhand des Gradienten bestimmt wird, bis ein Minimum gefunden wurde. Abhängig vom Startpunkt der Optimierung werden nur lokale Optima gefunden [42, S. 184]. Die Initialisierung der Gewichte ist deshalb und aufgrund der verwendeten Lernverfahren wichtig. Gleich initialisierte Gewichte führen zum Beispiel auch zu gleichmäßiger Veränderung dieser während des Lernens [42, S. 190]. Besser werden zufällige Startwerte verwendet, welche im Mittelwert bei null liegen, da das Training so schnell und effizient abläuft [42, S. 229] [36, S. 206]. Hier kommen mehrere Optionen infrage, welche entweder gleichmäßige (*uniform*) oder normalverteilte (*normal*) Zufallszahlen um 0 als Initialgewichte verwenden. Die Strategien heißen zum Beispiel *Random Uniform* und *Random Normal*. Weiterentwickelte Ansätze sind *Glorot*- bzw. *Xavier-Uniform* oder *-Normal* nach Xavier Glorot [49] sowie die für die ReLU-Aktivierungsfunktion entwickelten *He*- bzw. *Kaiming-Uniform* oder *-Normal* nach Kaiming He [58]. [42, S. 230 f.]

Nach der Initialisierung der Gewichte wird iterativ ein Gradientenverfahren angewandt. Dieser Prozess wird auch *Stochastic Gradient Descent* (SGD) genannt [42, S. 189]. Die Anzahl der Iterationen oder Durchläufe des Gradientenabstiegs, der Epochen [42, S. 222], muss vor dem Training festgelegt werden. Das Training kann dabei des Weiteren in sogenannte *Batches* aufgeteilt werden. Ein *Batch* bezeichnet die Anzahl der Datenpunkte, die den Trainingsdaten in einem Lernvorgang entnommen werden. Es wird also nicht immer die Ableitung und der Fehler der gesamten Trainingsmenge berechnet, sondern dieser Prozess wird nur für einen Teil der Daten durchgeführt. Die Gewichte werden nach jedem Lernvorgang mit einem *Batch* aktualisiert. Dieses Vorgehen wird als *Batch-Learning* bezeichnet [42, S. 188]. Als Standardwert für die *Batch*-Größe ist beispielsweise in *Keras* 32 festgelegt [42, S. 237]. Das Gradientenabstiegsverfahren wird also jeweils nur auf 32 Datenpunkte gleichzeitig angewandt. Größere oder kleinere *Batches* können ebenfalls eingestellt und getestet werden. Besteht eine Trainingsmenge nun zum Beispiel aus 320 Datenpunkten, wird der Gradientenabstieg pro Epoche 10-mal mit 32 Datenpunkten durchgeführt. Eine Epoche gilt somit als abgeschlossen, wenn die gesamte Trainingsmenge das Optimierungsverfahren durchlaufen hat. Weiterhin wichtig ist die Lernrate, welche die Stärke der Gewichtsänderung während des Lernens bestimmt [72, S. 25]. Wird die Lernrate zu klein gewählt, dauert der Lernprozess sehr lange, während eine zu große Lernrate das Finden eines Minimums verhindern kann, da dieses zum Beispiel übersprungen wird oder es zu Oszillationen um das Minimum kommt [72, S. 67].

Als Optimierer stehen wieder einige Optionen zur Auswahl wie zum Beispiel das hier erwähnte Verfahren SGD. Auch Weiterentwicklungen dieses Verfahrens wie *Adagrad*, *RMSProp* oder *Adam* werden in [42, S. 244 ff.] im Detail beschrieben und verwenden im Gegensatz zu SGD nicht konstante sondern adaptive Parameter der Optimierung, wie zum Beispiel eine adaptive Schrittweite (*learning rate*). Nach [42, S. 222, 246] ist der Algorithmus *Adaptive Moment Estimation* (*Adam*) oft eine gute Wahl.

Das Problem der Überanpassung wurde bereits angesprochen und tritt besonders bei Netzen mit vielen Neuronen in den versteckten Schichten auf [72, S. 77]. Die Anzahl der Schichten und Neuronen sollte also nicht zu groß gewählt werden. Das Problem kann durch die sogenannte Regularisierung auch in der Zielfunktion der Optimierung berücksichtigt werden, welche im Detail in [42, S. 255 ff.] erklärt wird. Im Grunde wird dabei die Modellkomplexität als numerisches Maß mit in die Zielfunktion aufgenommen. Es wird also versucht, ein Modell zu finden, welches die Fehler der Trainingsdaten minimiert, während auch die Modellkomplexität beispielsweise durch die Wahl geringer Gewichte möglichst klein bleibt. Das Problem der besonders genauen Abbildung der Trainingsdaten, der Überanpassung, kann somit verhindert werden. [42, S. 255 ff.]

Convolutional und Recurrent Neural Networks

Hier seien noch die Ansätze der *Convolutional Neural Networks* (CNN) und *Recurrent Neural Networks* (RNN) genannt. Dabei handelt es sich um Erweiterungen der herkömmlichen *Feed-forward* neuronalen Netze. In dieser Arbeit werden die Verfahren nicht verwendet, können aber in weiteren Forschungsvorhaben bezüglich ihrer Anwendung auf die Modellierung von Fernwärmenetzen untersucht werden. Dadurch lässt sich möglicherweise die Genauigkeit steigern. Ebenso kann mit den Ansätzen die thermisch instationäre Modellierung im Detail untersucht werden.

2.5.5. Ensemble Learning

Unter *Ensemble Learning* werden die beiden Ansätze des *Boosting* und *Bootstrap-Aggregating* (*Bagging*) zusammengefasst. Beim *Boosting* werden mehrere schwache Lernalgorithmen hintereinandergeschaltet [42, S. 342 ff.]. Ein Ansatz ist das *Gradient Boosting*, welches auf Entscheidungsbäumen basiert. Hier werden mehrere Bäume mit wenigen Verzweigungen sequenziell für das Lernen verwendet. Ausgehend vom ersten Baum wird das Ergebnis geschätzt und dieses kontinuierlich von den nachgeschalteten Bäumen verbessert. Ein aktueller Algorithmus dieses Ansatzes ist zum Beispiel *eXtreme Gradient Boosting* (XGBoost) [26].

Beim *Bagging* werden mehrere Datensätze durch Auswahl aus der Trainingsmenge mit Zurücklegen erstellt [42, S. 330]. Einige Datenpunkte sind also in den entstehenden Datensätzen mehrfach vorhanden. Die Datensätze werden dann von unterschiedlichen Lernalgorithmen verwendet und deren Ergebnisse gemittelt. Ein bekannter Ansatz ist die Methode *Random Forest*, welche wieder auf Entscheidungsbäumen als Lernalgorithmen basiert [42, S. 330].

Ensemble Learning Ansätze sind vor allem für das hier verwendete strukturierte Lernen interessant. Dort zeigen zum Beispiel die Algorithmen XGBoost und *Random Forest* neben tiefen neuronalen Netzen hohe Genauigkeiten [42, S. 28].

2.5.6. Bestärkendes Lernen

Bestärkendes Lernen wurde bereits eingangs dieses Kapitels erwähnt und bezeichnet das Lernen ohne ein bekanntes optimales Ergebnis des Algorithmus. Die Methode wird in [36, S. 313 ff.] und [42, S. 481 ff.] erläutert und soll hier nur grob beschrieben werden. Bestärkendes Lernen könnte direkt für die Regelung eines Fernwärmenetzes eingesetzt werden. In diesem Ansatz würde ein Netzmodell (physikalisch oder neuronales Netz) mit einem Reinforcement-Learning-Algorithmus gekoppelt werden, welcher die Bedienung des Fernwärmenetzes anhand des Modells erlernen soll. Als Algorithmus wird häufig ein weiteres neuronales Netz genutzt. Für das Training muss eine übergeordnete Reinforcement-Learning-Struktur geschaffen werden, die das automatisierte Training des neuronalen Netzes leitet, Lernsituationen schafft und richtige Steuereingriffe mit positivem Feedback sowie falsche

Eingriffe mit negativem Feedback bewertet. Das zu trainierende neuronale Netz kann entsprechend lernen, wie ein Fernwärmenetz optimal zu steuern ist.

3. Stand der Technik

In diesem Kapitel wird der Stand der Technik der Modellierung, des Betriebs und der Optimierung sowie Regelung von Fernwärmenetzen dargelegt. Kapitel 3.1 beschreibt die Simulationsgrundlagen sowie aktuelle Entwicklungen und Methoden im Bereich der Modellierung. Die hier angewandte Modellierung wird dabei erst im Kapitel 5 im Detail beschrieben. Im Folgenden wird in Kapitel 3.2 der Betrieb von Fernwärmenetzen beschrieben. Dabei wird auf konventionelle Betriebsweisen eingegangen, welche auch heute noch häufig angewandt werden. Anschließend wird in Kapitel 3.3.1 das Optimierungspotenzial von Fernwärmenetzen bezüglich derzeitiger Betriebsweisen recherchiert. Kapitel 3.3.2 beschreibt zusätzlich aktuelle Forschungsprojekte im Bereich der Optimierung und Regelung, welche bereits fortgeschrittene Betriebsweisen von Fernwärmenetzen entwickeln. Insgesamt soll dieses Kapitel eine Grundlage für anschließende Entscheidungen bezüglich der Modellierung, Optimierung und Regelung bilden.

3.1. Modellierung von Fernwärmenetzen

Im Folgenden wird auf die Grundlagen der Modellierung von Fernwärmenetzen sowie aktuell vorhandene Modelle und Entwicklungen eingegangen. Dieses Kapitel soll als Recherche für eine spätere Entscheidung bezüglich der Modellierungsansätze dienen. Die konkrete physikalische Beschreibung der Netze erfolgt in Kapitel 5.

Berechnungsmodelle von Fernwärmenetzen lassen sich nach Icking [61, S. 5] in verschiedene Kategorien einteilen. In der vorliegenden Arbeit sollen die hydraulischen und thermischen Zustände im Netz berechnet und als Zielgrößen oder Nebenbedingungen (Restriktionen) einer MPC verwendet werden. Die Berechnung der hydraulischen Größen Massenstrom und Druck bildet dabei eine Modellkategorie, die Berechnung der thermischen Größen wie Temperaturen und Wärmeverluste die zweite. Beide Rechnungen können dabei nicht in einem Schritt, sondern müssen getrennt gelöst werden. Da sich hydraulische und thermische Größen gegenseitig beeinflussen, werden hydraulische und thermische Gleichungen iterativ alternierend gelöst [61, S. 62].

Des Weiteren können Modelle stationär oder instationär berechnet werden. Bei stationären Simulationen wird davon ausgegangen, dass sich Zustände instantan im Netz ausbreiten. Es wird also ein eingeschwungener Zustand gesucht, an welchem sich sämtliche Drücke und Temperaturen vollständig von den Wärmeerzeugungsanlagen bis zu den Wärmeübergabestationen ausgebreitet haben. Stationäre Modelle von Fernwärmenetzen können für einen Zeitpunkt gelöst werden, um zum Beispiel Maximallastfälle für die Auslegung zu berechnen. Ebenfalls können Zeitreihen simuliert werden, um beispielsweise eine Prognose eines Tagesverlaufs zu erstellen oder einen solchen nachzurechnen. Diese Art der Aneinanderreihung von stationären Simulationen wird in dieser Arbeit auch quasi-stationäre Berechnung genannt. Da alle Zustände im Netz nur von den Eingangsgrößen im jeweiligen Zeitschritt abhängig sind, können mehrere Zeitschritte getrennt berechnet werden. Bei einer Optimierung von Einspeiseleistungen und Vorlauftemperaturen muss beachtet werden, dass mit stationären Berechnungen die Temperaturausbreitung im Netz vernachlässigt wird.

Bei instationären Berechnungen werden Druck- und Temperaturfronten hingegen berücksichtigt. Die Dynamik des Fernwärmenetzes wird abgebildet und die thermo-hydraulischen Zustände sind zeitabhängig. Eine getrennte Berechnung von mehreren Zeitschritten ist nicht mehr möglich. Druckfronten breiten sich im Netz in etwa mit der Schallgeschwindigkeit des Strömungsmediums aus. Temperaturen strömen näherungsweise mit dem Medium selbst durch das Netz und breiten sich so mit der Strömungsgeschwindigkeit aus. Für hydraulisch instationäre Berechnungen sind feine Zeitauflösungen im Sekunden- oder Millisekundenbereich erforderlich, während für thermisch instationäre Berechnungen in der Regel Simulationen im Minutenbereich ausreichen. [61, S. 6]

Aufgrund des erheblichen zeitlichen Aufwands für voll-instationäre Berechnungen kann für hydraulische Gleichungen eine stationäre und für thermische Gleichungen eine instationäre Betrachtung gewählt werden. Somit werden Druckwellen vernachlässigt. Diese Art der Simulation eignet sich also nicht uneingeschränkt für die Auslegung von Druckregelungen. Ein weiteres Problem bei der instationären Betrachtung ist die Kenntnis der Anfangszustände. Dynamische Übergangszustände sind nicht immer bekannt, für die Initialisierung der Simulation sind allerdings bekannte Werte notwendig. So können stationäre Simulationen für die Berechnung von Anfangswerten herangezogen werden [61, S. 5]. Bei bekannter Temperaturverteilung im Netz kann auch diese für die Initialisierung der thermisch instationären Rechnungen verwendet, während die hydraulischen Zustände anschließend stationär bestimmt werden.

Letztendlich kommen verschiedene Berechnungsmethoden und verfügbare Implementierungen für die Modellierung von Fernwärmenetzen infrage. Detaillierte Zusammenfassungen zu diesem Thema finden sich in den Studien [7], [94] und [121]. Die für diese Arbeit relevanten Verfahren und Implementierungen werden im Folgenden vorgestellt.

3.1.1. Physikalische Netzmodelle

Fernwärmenetzmodelle können als sogenannte Knoten-Kanten-Modelle realisiert werden, welche auf der Graphentheorie basieren. In diesen Modellen wird der Zusammenhang zwischen den Rohrleitungen im Netz, Knotenpunkten und anderen Komponenten wie Pumpen oder Ventilen in einer Knoten-Kanten-Matrix dargestellt. Die Berechnung der hydraulischen und thermischen Gleichungen kann anschließend durch das Lösen linearer Gleichungssysteme erfolgen. Ein Beispiel eines solchen Modells ist das Matrix-Modell nach Strelow [119].

Ein wichtiger Aspekt bei der Modellierung von Fernwärmenetzen ist die Ermittlung des Wärmeverbrauchs. Dieser beeinflusst maßgeblich die hydraulischen und thermischen Zustände sowie die notwendige Wärmeeinspeisung in das Netz und ist somit als Eingangswert für eine Simulation erforderlich. Bei Nachrechnungen ist eine Bestimmung des Wärmeverbrauchs aus Messwerten möglich, während bei Prognoserechnungen auch eine Prognose des Wärmebedarfs durchgeführt werden muss. Eine Software für Prognosen und wirtschaftliche MILP-Optimierungen ist zu Beginn des in dieser Arbeit beschriebenen Projektes bereits vorhanden. Auf Entwicklungen im Bereich der Prognoseverfahren wird daher nicht weiter eingegangen und eine gute Prognosegenauigkeit durch entsprechende Software angenommen. Kommerziell steht für diese Aufgabe zum Beispiel die Software *BoFiT* zur Verfügung, in welcher ursprünglich auch Knoten-Kanten-Modelle für die stationäre und instationäre Berechnung von Fernwärmenetzen entwickelt wurden. Im Rahmen des *Verbundprojekts: Bessere Ausnutzung von Fernwärmeanlagen* wurde zum Beispiel die Arbeit von Icking [61] veröffentlicht, in welcher die Knoten- und Maschenverfahren vorgestellt und durch Druckregelungen ergänzt werden. Des Weiteren wird eine Methode zur Lösung der thermischen Zustände in einem Gleichungssystem entwickelt, welche es ermöglicht, vermaschte Fernwärmenetze zu berechnen. Thermisch stationäre sowie instationäre Berechnungsmethoden werden vorgestellt. In [68] wird ausgenutzt, dass die Knoten-Kanten-Matrizen

von Fernwärmenetzen dünn besetzt sind. Die Berechnung kann so durch die Verwendung von *sparse*-Matrizen beschleunigt werden. Weitere Zeitersparnis ergibt sich durch die Verwendung analytisch berechneter Ableitungen für die Lösung der Gleichungssysteme. Anschließend befassen sich einige Arbeiten mit der Verbesserung der thermo-hydraulischen Modelle für einen Einsatz in Optimierungen. In Löwen [79] wurde eine Methode entwickelt, Modelle von Fernwärmenetzen zu aggregieren und somit die Anzahl der Gleichungen zu reduzieren. Die Methode erlaubt eine akkurate Berechnung der Drücke im Netz bis zu einem bestimmten Aggregationsgrad bei gleichzeitig deutlich verringerten Simulationszeiten. In Larsen [75] wird diese Aggregationsmethode mit einer weiteren dänischen Methode verglichen, welche sich besser für die Aggregation unter Einhaltung der Wärmeverluste eignet. Die Beibehaltung korrekter Druckverluste wird allerdings nicht beachtet. Eine aktuellere Methode der Vereinfachung und Aggregation von Fernwärmenetzmodellen wird von Bott [18] vorgestellt. Hier wird unter anderem basierend auf den Aggregationsmethoden von Löwen [79] und Larsen [75] eine teilweise Aggregation der Netzmodelle vorgenommen und durch eine Vereinfachung der Druckverlustgleichungen ergänzt.

Knoten-Kanten-Modelle können theoretisch in vielen Programmiersprachen implementiert werden. In dem Projekt [100] wird zum Beispiel ein stationäres hydraulisches Modell in *C++* erstellt und für die Detektion von Leckagen in Fernwärmenetzen verwendet. Oft wird auch *MATLAB*[®] für die Modellierung eingesetzt, wie in einigen Projekten in Kapitel 3.3.2 beschrieben wird. Auch grafische Oberflächen wie *Simulink*[®] für *MATLAB*[®] oder *Dymola* für *Modelica* können verwendet werden. In diesem Fall werden Fernwärmenetzmodelle mittels grafischer Blöcke für die verschiedenen Komponenten erstellt und sind so direkt grafisch dargestellt. Ein *Simulink*[®] Modell wurde bereits in einem Projekt am Institut für Automatisierungstechnik der Universität Bremen erstellt. In einer Projektarbeit wurde ein thermisches und hydraulisches Modell gebildet, welches anschließend für den Entwurf eines hydraulischen Regelungskonzeptes verwendet wird.

Für *Modelica* wurden einige Bibliotheken entwickelt, welche die Modellierung vereinfachen und zum Beispiel den Import von sogenannten GIS-Daten (GeoInformationSystem) ermöglichen. Diese Daten sind für die verlegten Rohre in der Regel bereits vorhanden und geben deren genaue Standorte im Netz an [113, S. 163]. So kann ein Modell des Netzaufbaus automatisiert erzeugt werden. Ebenfalls sind bereits vorgefertigte Modelle für die verschiedenen Komponenten von Fernwärmenetzen vorhanden. Zu nennen sind hier die Projekte [56], [31] und [113]. In [56] werden *Modelica*-Modelle programmiert, um größere Netzwerke darstellen zu können. Ebenso wird ein Ansatz vorgestellt, Netzwerkdaten zu aggregieren und direkt in *Modelica*-Modelle zu konvertieren. [31] entwickelt detaillierte Modelle im Rahmen des *AMBASSADOR*-Frameworks der Europäischen Union. Bei der Modellierung soll dabei spezifisch darauf geachtet werden, dass die Modelle schnell und akkurat genug für den Einsatz in MPCs sind. Die entwickelten Modelle werden auf ein kleines Fernwärmenetz angewandt und notwendige Verbesserungen für ein Folgeprojekt vorgeschlagen. In [113] wird ein Framework vorgestellt, welches den Import von GIS-Daten ermöglicht und mehrere Komponentenmodelle enthält. Ein Literaturreview über Projekte, die *Modelica* verwenden, wird in [6] präsentiert.

Die Vorteile der Knoten-Kanten-Modelle, grafischen Oberflächen und einfachen Möglichkeiten der Netzerstellung und -aggregation werden in einigen kommerziellen Softwaretools vereinigt. Als Beispiel sollen *STANET*[®] vom Ingenieurbüro Fischer-Uhrig und *SIR 3S*[®] von der 3S Consult GmbH genannt werden. Beide Programme können für die Berechnung von Versorgungsnetzen verwendet werden und bieten die Möglichkeit, Netze durch den Import von GIS-Daten zu erstellen. Modelle können hydraulisch und thermisch stationär sowie instationär berechnet werden. Ebenso sind Algorithmen zur Aggregation der Netze integriert. Teile der in dieser Arbeit untersuchten Fernwärmenetze wurden in den Programmen bereits modelliert und analysiert.

3.1.2. Modellierung durch Ansätze des maschinellen Lernens

Neben den bisher vorgestellten physikalischen Modellierungsansätzen sollen des Weiteren die Verfahren Systemidentifikation und maschinelles Lernen genannt werden. Mithilfe dieser können Fernwärmemodelle basierend auf Messdaten oder Simulationsdaten aus physikalischen Modellen approximiert werden. Die Theorie der Ansätze und deren Algorithmen wurde in den Kapiteln 2.4 und 2.5 vorgestellt und bildet eine Grundlage für die spätere Auswahl der Modellierungsansätze. Von den Ansätzen wird eine deutliche Reduktion der Rechenzeit bei ähnlicher oder zumindest ausreichender Genauigkeit erwartet. Daher wird hier kurz auf Projekte eingegangen, welche bereits Ansätze des maschinellen Lernens für die Modellierung von Fernwärmenetzen einsetzen. Einige Projekte verwenden zum Beispiel neuronale Netze. In [67] wird mittels neuronaler Netze eine Exergieanalyse des Fernwärmenetzes in Afyonkarahisar durchgeführt. Für die Analyse werden Temperaturdaten von vier Jahren ausgewertet und ein neuronales Netz in *MATLAB*[®] erstellt. [66] erweitert das Projekt um Analysen für eine verbesserte und wirtschaftlichere Operation des Netzes. In [84] wird ein neuronales Netz basierend auf Daten von verschiedenen Messstationen in einem Fernwärmenetz trainiert. Ziel ist es, einige der Stationen durch Vorhersagen zu ersetzen. [73] stellt eine Möglichkeit der Modellvereinfachung durch maschinelles Lernen vor. Ein simples stationäres Rohrleitungsmodell für den Wärmeverlust wird basierend auf Lerndaten eines komplexeren instationären Modells erweitert. Die Simulationszeit des erweiterten simplen Modells ist dabei deutlich reduziert gegenüber dem komplexen Modell. Dynamische Effekte können aber berücksichtigt werden. In [5] wird zunächst ein Modell eines solarbetriebenen Fernwärmenetzes in der Software *TRNSYS* erstellt und anschließend durch ein neuronales Netz approximiert. Die Parameter des neuronalen Netzes werden vorab durch *Bayes'sche* Optimierung festgelegt.

3.2. Betrieb von Fernwärmenetzen

In diesem Kapitel wird der Betrieb von Fernwärmenetzen erläutert. Dazu zählen die relevanten Regelgrößen und wie die Versorgung sämtlicher angeschlossener Verbraucher sichergestellt wird. Dieses Kapitel orientiert sich in großen Teilen an Kapitel 3 der 1995 veröffentlichten Arbeit von Icking [61] und stellt somit herkömmliche Betriebsweisen dar, welche oft noch heute verwendet werden. Der Stand der Technik bezüglich des weiteren Optimierungspotenzials und dafür benötigter Regelungsalgorithmen wird im folgenden Kapitel 3.3 herausgestellt.

Der Betrieb und die Regelung von Fernwärmenetzen kann in zwei Ebenen unterteilt werden. Zum einen ist zu jeder Zeit ein bestimmter Wärmebedarf im Netz vorhanden, welcher von den Wärmeerzeugungsanlagen gedeckt werden muss. Dafür kann die benötigte Einspeiseleistung auf verschiedene Anlagen verteilt und mit bestimmten Vorlauftemperaturen eingespeist werden. Für diese Größen müssen Sollwerte generiert und anschließend ins Netz übertragen werden. Zum anderen werden Druck, Massenstrom und Temperatur so geregelt, dass die vorgegebenen Sollwerte eingehalten werden können.

Fernwärmenetze werden für die Versorgung von Haushalten und Industrie sowohl mit Warmwasser als auch mit Raumwärme verwendet. Der Warmwasseranteil ist über ein Jahr relativ konstant, während der Raumwärmebedarf schwankt. In Sommermonaten ist keine oder kaum Raumwärme nötig, während nur an wenigen Tagen im Winter die volle Leistung eines Fernwärmenetzes und dessen Erzeugungsanlagen abgerufen wird. So werden zum Beispiel Leistungen über 80 % des Maximalwertes in weniger als 10 % der Nutzungsstunden abgerufen [61, S. 14]. Für diese Maximalleistungen müssen das Fernwärmenetz und der Erzeugerpark allerdings ausgelegt werden. Die Temperaturen und Leistungen der Wärmeerzeugungsanlagen und Pumpen sowie die Rohrleitungen müssen entsprechend dimensioniert werden. Der über das Jahr schwankende Wärmebedarf lässt sich in Jahresdauerlinien darstellen, wie beispielhaft in Abbildung 3.1. In der Jahresdauerlinie ist die Wärmeleistung in Prozent

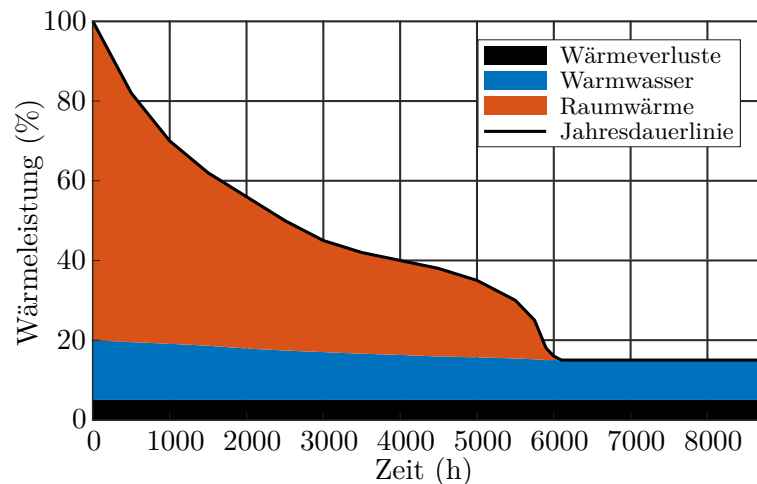


Abbildung 3.1.: Jahresdauerlinie eines Fernwärmenetzes nach [92, S. 18].

über der Nutzungsdauer in Stunden aufgetragen. In der Abbildung ist ebenfalls grob die Aufteilung der Einspeiseleistung in Wärmeverluste, Warmwasser und Raumwärme dargestellt.

Auch über den Tagesverlauf schwankt der Wärmebedarf in Fernwärmenetzen stark. Warmwasser wird nachts kaum benötigt und Heizungen häufig abgesenkt. Tagsüber treten Lastspitzen auf und hohe Wärmeleistungen werden abgerufen. Auch durch Wetterveränderungen können plötzliche Schwankungen im Betrieb auftreten, da Raumheizungen oft Außentemperatur-geregelt werden [61, S. 15]. Die Leistungen der Wärmeerzeugungsanlagen müssen permanent angepasst werden. Der Wärmestrom \dot{Q} setzt sich nach Gleichung (3.1) aus der Differenz zwischen Vorlauftemperatur T_{VL} und Rücklauftemperatur T_{RL} , dem Massenstrom \dot{m} und der Wärmekapazität des Mediums c_p zusammen [61, S. 23].

$$\dot{Q} = \dot{m} \cdot c_p \cdot (T_{VL} - T_{RL}) \quad (3.1)$$

Um die Leistung einer Anlage zu erhöhen, kann entweder die Vorlauftemperatur erhöht oder der Massenstrom durch die Umwälzpumpen gesteigert werden. Mit höheren Massenströmen steigen auch die Druckverluste im Netz und die benötigte Leistung der Pumpen steigt stark an. Somit wird ein erhöhter Wärmebedarf nicht nur durch eine Anhebung des Massenstroms abgefangen, auch die Vorlauftemperaturen können erhöht werden. Erhöhte Vorlauftemperaturen führen durch den höheren Temperaturunterschied zwischen Medium, Rohren und Umgebung aber zu größeren Wärmeverlusten [92, S. 11]. Die Vorlauftemperaturen werden dementsprechend nicht beliebig gesteigert. Eine Möglichkeit der Festlegung der Vorlauftemperatur ist eine außentemperaturabhängige Vorlauftemperaturkurve, welche im Vorfeld heuristisch oder durch Analysen bestimmt und anschließend im Betrieb angewandt wird. Eine weitere Möglichkeit ist die Anpassung der Vorlauftemperatur an Wärmebedarfsprognosen und den tatsächlich anliegenden Wärmebedarf. Wärmebedarfsprognosen basieren ebenfalls hauptsächlich auf Außentemperaturprognosen. Weitere Faktoren wie Niederschlag, Windgeschwindigkeiten oder die Sonnenstrahlung können mit einbezogen werden [61, S. 15]. Da sich die Temperaturen nur etwa mit der Fließgeschwindigkeit des Mediums im Netz ausbreiten, macht es zusätzlich Sinn, Temperaturen zum Beispiel morgens schon vor einer Lastspitze anzuheben. Bei prognostiziertem geringem Wärmebedarf können die Temperaturen dann frühzeitig wieder abgesenkt werden. [61, S. 15]

Der Wärmestrom wird bei festgelegter Vorlauftemperatur von den Pumpen durch den Massenstrom geregelt. Dabei kann eine Differenzdruckregelung eingesetzt werden. In einer Wärmeübergabestation wird, wie in Kapitel 2.1.5 beschrieben, ein bestimmter Differenzdruck benötigt. Sobald die eingespeiste Leistung nicht mehr dem Wärmebedarf im Netz entspricht, fällt der Differenzdruck an den Wärmeübergabestationen ab und eine Unterversorgung tritt auf. Das Netz muss also so geregelt werden, dass an jeder Wärmeübergabestation ein ausreichender Differenzdruck vorhanden ist. Für die Diffe-

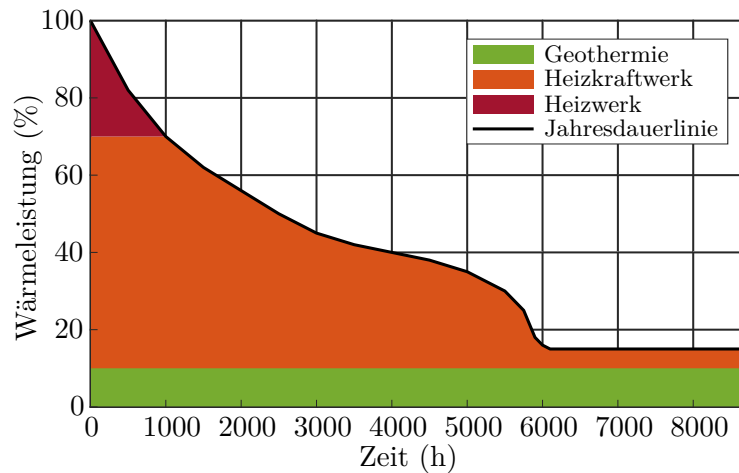


Abbildung 3.2.: Jahresdauerlinie eines Fernwärmenetzes mit Verteilung der Wärmeleistung auf verschiedene Wärmeerzeugungsanlagen.

renzdruckregelung werden häufig Referenzmessstellen an oder in der Nähe von Netzschlechtepunkten verwendet [92, S. 55]. Bei mehreren Netzschlechtepunkten wird jeweils die Messstelle mit dem kleinsten Differenzdruck für die Regelung verwendet. Unterschreitet nun ein Schlechtepunkt den Solldifferenzdruck, werden die Drehzahlen der Pumpen erhöht und der Massenstrom steigt. Der Druck breitet sich dabei mit der Schallgeschwindigkeit im Netz aus. Differenzdrücke können durch die Massenstromregelung der Pumpen somit im Vergleich zu einer Vorlauf-temperaturregelung schnell angepasst werden.

Durch den schwankenden Wärmebedarf über den Tages- und Jahresverlauf kann unter Umständen nicht der ganze Wärmebedarf von einer zentralen Anlage gedeckt werden. Zusätzlich zu zentralen KWK-Anlagen werden dezentrale Heizwerke eingesetzt. Sind mehrere Wärmeerzeugungsanlagen in einem Netz vorhanden, wird der Differenzdruck nur durch eine Anlage beziehungsweise einen Pumpensatz geregelt. Diese Anlage speist einen variablen Massenstrom und eine variable Leistung ein. Die anderen Anlagen speisen festgelegte Leistungen oder Massenströme bei einer vorzugebenden Vorlauf-temperatur ein. [92, S. 55]

Die verschiedenen Wärmeerzeugungsanlagen können in drei Kategorien eingeteilt werden: Grund-, Mittel- und Spitzenlastanlagen. Die Unterteilung erfolgt hauptsächlich nach den Erzeugungskosten und der Verfügbarkeit. Grundlastanlagen sind zum Beispiel Abwärme- oder Müllverbrennungsanlagen, welche relativ konstant Wärme liefern. Auch regenerative Energiequellen wie Geothermie können in Zukunft als Grundlastanlagen verwendet werden. Als Mittellastanlagen werden KWK-Anlagen verwendet. Dabei ist darauf zu achten, dass die Wärmeproduktion auch die Stromproduktion beeinflusst und der Betrieb somit abgestimmt werden muss. Spitzenlastanlagen sind Anlagen, welche bei hohen Lasten möglichst schnell angefahren werden können. Hier werden Heizwerke eingesetzt, in welchen die Wärmeproduktion im Vergleich teuer ist. [61, S. 16]

Für die Verteilung der Last auf die Wärmeerzeugungsanlagen werden häufig Einsatzpläne für das Betriebspersonal erstellt [61, S. 16]. Diese können zum Beispiel festlegen, ab welcher Wärmelast welche Anlage einzusetzen ist. Diese Pläne können generell festgelegt, wie beispielhaft mit der Jahresdauerlinie in Abbildung 3.2, oder für jeden Tag einzeln bestimmt werden. Eine kontinuierliche Optimierung ist ebenfalls möglich, erhöht die Flexibilität des Anlageneinsatzes und wird in einigen Fernwärmenetzen schon durchgeführt. Aus den beschriebenen Gegebenheiten ergibt sich ein Optimierungspotenzial der einzuspeisenden Wärmeleistungen der verschiedenen Wärmeerzeugungsanlagen, den Vorlauf-temperaturen und den Pumpenleistungen. Bei der Optimierung können verschiedene Faktoren wie Kosten, Emissionen oder physikalische Restriktionen berücksichtigt werden. Diese Thematik wird detailliert im folgenden Kapitel 3.3 behandelt.

3.3. Optimierung und Regelung von Fernwärmenetzen

In den vorherigen Kapiteln wurden die Modellierung und der Betrieb von Fernwärmenetzen beschrieben. Zudem wurden in den Kapiteln 2.2 und 2.3 Grundlagen der Optimierung und Modellprädiktiver Regelungen vorgestellt. Auf dieser Basis soll im Folgenden das zentrale Thema der optimalen Regelung verteilter Wärmeinspeisung in Fernwärmenetzen erarbeitet werden. Zunächst wird in Kapitel 3.3.1 festgestellt, wie Fernwärmenetze optimal betrieben werden können. Dazu zählen die Größen, welche die Effizienz von Fernwärmenetzen beeinflussen und somit sinnvoll optimiert werden können, um einen möglichst günstigen Betrieb zu ermöglichen. Anschließend werden in Kapitel 3.3.2 relevante Projekte zur Optimierung und Regelung von Fernwärmenetzen vorgestellt.

3.3.1. Optimale Fahrweise von Fernwärmenetzen

Der Betrieb von Fernwärmenetzen wurde in Kapitel 3.2 zusammengefasst. Insgesamt muss der Wärmebedarf im Netz gedeckt werden. Dafür müssen die Einspeiseleistung und Vorlauftemperaturen der einzelnen Wärmeerzeugungsanlagen bestimmt werden. Hier kann eine Optimierung nach verschiedenen Kriterien erfolgen. Anschließend werden diese Sollwerte durch Regelungen umgesetzt. Dabei kommen hauptsächlich die lokalen Temperatur-, Druck und Massenstromregelungen zum Einsatz. Diese Regelungen sorgen zusammen dafür, dass der Wärmebedarf aller Wärmeübergabestationen gedeckt wird.

Zunächst stellt sich also die Frage, wie der benötigte Wärmebedarf optimal auf verschiedene Wärmeerzeugungsanlagen verteilt werden kann. Für diese Aufgabe werden die sogenannten *Unit-Commitment*- und *Economic-Dispatch*-Probleme gelöst, welche zum Beispiel in [28] beschrieben und aus der Kraftwerkseinsatzoptimierung in der Stromproduktion stammen.

Unit-Commitment- und Economic-Dispatch-Problem

Dabei handelt es sich um mathematische Optimierungsprobleme, welche Einsatzpläne mehrerer Erzeuger für die Stromproduktion generieren. Die Einsatzpläne werden für einen bestimmten Zeithorizont in diskreten Schritten ermittelt und unterliegen gegebenen Restriktionen. Das *Unit-Commitment*-Problem wird herkömmlicherweise als gemischt ganzzahliges lineares Programm (GGLP) definiert, welches die Kosten der Stromerzeugung minimieren oder den Profit maximieren soll [86]. Durch das Lösen des *Unit-Commitment*-Problems wird für jeden Schritt definiert, welche Erzeugungsanlagen ein- oder ausgeschaltet sein sollen. Im nächsten Schritt, dem *Economic-Dispatch*-Problem, wird für jede Anlage die zu erzeugende Leistung berechnet. Die beiden Probleme sind ähnlich formuliert auf die Wärmeproduktion in Fernwärmenetzen anwendbar. Dabei werden bisher ebenfalls vorrangig die Ziele der Kostenminimierung oder Profitmaximierung verwendet. Theoretisch können weitere Ziele wie die Minimierung von Emissionen oder die maximale Ausnutzung bestimmter Anlagen hinzugezogen werden. Bei bekanntem Wärmebedarf sind für die Optimierung entsprechend Informationen wie die Erzeugungskosten aller Anlagen, mögliche Profite sowie Emissionswerte notwendig.

Beim Einsatz von KWK-Anlagen in Fernwärmenetzen werden Strom- und Wärmeproduktion nicht getrennt optimiert. Je mehr Leistung einer solchen Anlage für die Wärmeproduktion verwendet wird, desto weniger steht für die Stromproduktion zur Verfügung. Entsprechend werden das *Unit-Commitment*- und *Economic-Dispatch*-Problem für Strom- und Wärmenetze in der Regel kombiniert gelöst, um den Bedarf in beiden Netzen zu decken [79, S. 3]. Ist ein Fernkältenetz am selben Standort vorhanden, kann auch dieses mit in die Optimierung einbezogen werden.

Durch die Definition der Optimierungsprobleme als lineares Programm können nicht-lineare physikalische Restriktionen der Fernwärmenetze nicht berücksichtigt werden. Allgemein ist es schwierig, eine

vollständige Optimierung von Fernwärmenetzen unter der Berücksichtigung von Wärmeerzeugungsanlagen mit möglichen An- und Aus-Zuständen sowie nicht-linearen Restriktionen in einem Schritt durchzuführen. Es ergibt sich ein gemischt ganzzahliges nicht-lineares Programm, wie in Kapitel 2.2 vorgestellt. Die Lösung solch einer Optimierung beinhaltet sowohl die Problematik der beschränkten nicht-linearen Optimierung als auch die der diskreten Optimierung mit Sprüngen in der Zielfunktion und den Nebenbedingungen. Für diese Art der Probleme gibt es derzeit keine effizienten Lösungsalgorithmen [126]. Der Fortschritt bezüglich der Lösung dieser Probleme mittels globaler Optimierung kann zum Beispiel [19] entnommen werden. Bezüglich der Anwendung auf Fernwärmesysteme wird stattdessen unter anderem in [106] vorgeschlagen, die *Unit-Commitment*- und *Economic-Dispatch*-Probleme in zwei Schritten zu lösen. Das *Unit-Commitment*-Problem wird dabei als gemischt ganzzahliges lineares Programm beschrieben, um den Status verschiedener Wärmeerzeugungsanlagen zu bestimmen. Das *Economic-Dispatch*-Problem wird als beschränkte nicht-lineare Optimierung gelöst.

Hier soll nun darauf eingegangen werden, welches weitere Optimierungspotenzial sich durch eine zusätzliche nicht-lineare Optimierung ergibt. Zunächst können nicht-lineare hydraulische und thermische Restriktionen des Fernwärmenetzes in der Optimierung berücksichtigt werden. Neben der wirtschaftlichen Optimierung ist so zu jedem Zeitpunkt sichergestellt, dass zum Beispiel Differenzdrücke, Minimaldrücke und Vorlauftemperaturen im Netz hoch genug sind. Die wirtschaftlich optimierten Sollwerte für Einspeiseleistungen und Vorlauftemperaturen werden also auch hinsichtlich ihrer Umsetzbarkeit geprüft. Weiterhin können theoretisch Vorlauftemperaturen genauer optimiert und Speichereffekte im Netz berücksichtigt werden.

Vorlauf- und Rücklauftemperaturoptimierung

Die Vorlauftemperatur hat einen maßgeblichen Einfluss auf die Wärmeverluste im Netz. Je geringer die Vorlauftemperatur, desto geringer sind auch die Temperaturunterschiede zwischen dem geführten Medium und der Außenumgebung. Zusätzlich steigt bei sinkender Vorlauftemperatur nach Gleichung (3.1) der benötigte Massenstrom im Netz, um den Wärmebedarf zu decken. Durch höhere Strömungsgeschwindigkeiten verringert sich der Wärmeverlust weiter. Niedrigere Vorlauftemperaturen führen in KWK-Anlagen dazu, dass mehr Elektrizität produziert werden kann [15, S. 23]. Des Weiteren können theoretisch Wärmequellen niedrigerer Temperatur wie zum Beispiel Geothermie implementiert werden. Die Temperatur kann aber nicht beliebig abgesenkt werden. Höhere Massenströme führen zu größeren Druckverlusten und der Leistungsbedarf der Pumpen steigt. Somit muss bei der Temperaturoptimierung zwischen Einsparungen durch Reduktion der Wärmeverluste und erhöhten Pumpkosten sowie Netzrestriktionen abgewogen werden.

Dazu kommt, dass die Vorlauftemperaturen auch einen Einfluss auf die Wärmeübergangsstationen und somit die Rücklauftemperaturen haben. Niedrigere Vorlauftemperaturen führen bei gleichbleibender Netzarchitektur und Verbrauchssituation durch höhere Massenströme zu höheren Rücklauftemperaturen [138, S. 227]. In Fernwärmenetzen entstehen nur etwa ein Drittel der Wärmeverluste im Rücklauf [138, S. 228]. Ein Anstieg der Wärmeverluste im Rücklauf könnte somit von größeren Einsparungen im Vorlauf kompensiert werden. Weitere Effekte sind allerdings zu beachten. So können höhere Rücklauftemperaturen einen Einfluss auf die Dampfauskopplung und somit auch auf die Stromproduktion in KWK-Anlagen haben [139, S. 23]. In [15] werden weitere Effekte einer Vorlauftemperaturabsenkung im Detail untersucht und verschiedene Temperaturkurven auf ein Fernwärmenetz angewandt. Es wird deutlich, dass eine Absenkung häufig durch die verwendeten Wärmeübergabestationen und deren Spezifikationen limitiert und aus ökonomischer Sicht nicht immer sinnvoll ist. Im untersuchten Fernwärmenetz wurde neben anderen Temperaturprofilen untersucht, die Vorlauftemperatur im Schnitt um $15\text{ }^{\circ}\text{C}$ abzusenken. Dadurch wurde der Massenstrom um 45 bis 50 % und die Rücklauftemperatur um $0,8\text{ }^{\circ}\text{C}$ erhöht.

Einige Projekte untersuchen die Absenkung der Rücklauftemperatur, um die Effizienz von Fernwärmenetzen zu steigern. [139] fasst die Vorteile niedrigerer Rücklauftemperaturen zusammen, welche im Wesentlichen das Gegenteil zu den bereits hier genannten Nachteilen bei erhöhter Rücklauftemperatur sind. [92, S. 50] beschreibt niedrigere Versorgungspreise bei niedrigerer Rücklauftemperatur als Anreiz, um Temperaturen zu reduzieren. Durch diese finanziellen Anreize können zum Beispiel Wärmeübergabestationen modernisiert werden und so bei gleichen Massenströmen eine höhere Temperaturabsenkung auf Fernwärmeseite ermöglichen. Eine weitere Möglichkeit besteht in der adaptiven Regelung der Vorlauftemperaturen auf Verbraucherseite, um auf Fernwärmenetzseite möglichst geringe Rücklauftemperaturen zu erreichen [138, S. 247]. Die Aktion liegt hier auf Kundenseite oder kann direkt von Stadtwerken durch Modernisierungsmaßnahmen der Netze durchgeführt werden. Im Netzbetrieb kann die Rücklauftemperatur im Gegensatz zur Vorlauftemperatur nicht direkt beeinflusst werden. Eine Optimierung der Vorlauftemperatur muss allerdings die Erhöhung der Rücklauftemperatur und damit einhergehende Nachteile beachten, um einen insgesamt optimalen Betrieb zu ermöglichen.

Bei der Änderung der Temperaturen sind zusätzlich maximale Temperaturgradienten einzuhalten. Bei starker Änderung der Temperaturen können Spannungen in den verlegten Rohren auftreten, welche zu vorzeitiger Alterung führen [74]. Ebenso sind die Temperaturen mehrerer Wärmeerzeugungsanlagen aufeinander abzustimmen.

Zusätzliche Sensorik im Fernwärmenetz verbessert möglicherweise Optimierungen und Regelungen. Die Thematik wird zum Beispiel in [14] untersucht, wird aber in dieser Arbeit nicht aufgegriffen.

Entsprechend der vorgestellten Problematik resultiert eine dynamische Veränderung der Vorlauftemperatur mit hohen Gradienten und dem Ziel der Temperaturabsenkung in komplexen Optimierungen. Im nächsten Kapitel werden einige solche Optimierungen vorgestellt, welche allerdings bisher kaum zur Anwendung in Fernwärmenetzen kommen.

Speichernutzung

Zum Einsatz kommen allerdings schon Optimierungen, welche versuchen die Wärmeeinspeisung besser über den Tag zu verteilen, um während Lastspitzen den Einsatz von zusätzlichen Heizwerken zu reduzieren. Diese Ansätze werden auch *peak-shaving* und *valley-filling* genannt. Infrage kommt der Einsatz thermischer Speicher, wie in Kapitel 2.1.1 beschrieben. Speicher können während Perioden geringer Last gefüllt und anschließend bei Lastspitzen entleert werden. So lässt sich die Maximalleistung einer Wärmeerzeugungsanlage erhöhen. Ebenso können Speicher bei günstigen Brennstoff- oder Strompreisen genutzt werden. Besonders für KWK-Anlagen ergibt sich ein zusätzliches Optimierungspotenzial, welches ein flexibleres Zusammenspiel der Strom- und Wärmeproduktion ermöglicht. Die Ansätze des *peak-shaving* und *valley-filling* ermöglichen konstantere Fahrweisen der Anlagen und somit auch eine weniger stark schwankende Stromproduktion. Wärmespeicher werden häufig in Wärmeerzeugungsanlagen errichtet, können aber auch dezentral im Netz verteilt werden [92, S. 40]. Für die Optimierung werden Speichermodelle in den in Kapitel 3.3.2 beschriebenen Projekten eingesetzt.

Auch das Volumen der Leitungen im Fernwärmenetz selbst kann als Speicher verwendet werden. Nachts kann zum Beispiel eine höhere Temperatur und Leistung in das Netz gefahren werden als notwendig. Die zusätzliche Leistung kann dann Lastspitzen am Morgen abfangen. Diese Art der Vorlauftemperaturoptimierung kommt bereits in einigen Fernwärmenetzen zur Anwendung und wird zum Beispiel in dem Projekt [80] anhand von *MATLAB*[®] / *Simulink*[®] Modellen kleiner Netze vorgestellt. Die Untersuchung des Speicherpotenzials von Fernwärmenetzen wird auch in [117] mit einem *MATLAB*[®] Modell basierend auf [51] durchgeführt. Dabei ist zu beachten, dass Wärme nicht verlustfrei gespeichert werden kann. Bei der Nutzung des Netzes als Speicher und vorzeitiger Vorlauftemperaturerhöhung wird zum Beispiel kurzfristig auch der Wärmeverlust erhöht.

[126] untersucht die Flexibilitätssteigerung von Fernwärmenetzen durch den Einsatz von Speichern und durch weitere Maßnahmen wie der fairen Verteilung von Wärme bei Knappheit. Auch [138, S. 241 ff.] beschreibt Ansätze, wie bei Spitzenlasten die Wärmeerzeugung durch Steuerung der Wärmeübergabestationen reduziert werden kann. Hier kann zum Beispiel eine Priorität auf die Warmwasserversorgung gelegt und Heizungen kurzzeitig ausgestellt oder heruntergeregelt werden. So besteht auch die Möglichkeit der Kommunikation zwischen Wärmeübergabestationen, um diese Absenkung möglichst auf viele Verbraucher zu verteilen. Optimierungen auf Verbraucherseite werden in dieser Arbeit nicht weiter behandelt. Die Netze sollen in gegebenen Ausbauzuständen nur im Betrieb optimiert werden.

3.3.2. Überblick relevanter Forschungsprojekte

In diesem Abschnitt werden relevante Forschungsprojekte im Bereich Optimierung und Regelung von Fernwärmenetzen vorgestellt. Die Projekte sollen dabei in drei Teile eingeteilt werden. Zunächst geht es um Optimierungen, welche hauptsächlich für die Generierung optimaler Sollwerte für Wärmeerzeugungsleistungen und Temperaturen entwickelt werden. Die Aufbringung der Sollwerte auf das Fernwärmenetz bleibt in diesen Veröffentlichungen offen und kann zum Beispiel manuell durch das Personal erfolgen. Anschließend werden MPCs vorgestellt. Diese beinhalten ebenfalls eine Optimierung der Sollwerte und sind um eine Regelung erweitert, welche ein Fernwärmenetz automatisieren könnte. Zuletzt werden Projekte vorgestellt, welche Algorithmen des maschinellen Lernens verwenden.

Basierend auf den Erkenntnissen dieses Kapitels soll im anschließenden Kapitel 4 ein Regelungskonzept für die vier in dieser Arbeit behandelten, zunehmend komplexen, vermaschten Fernwärmenetze erarbeitet werden. Das größte Fernwärmenetz wird mit Spitzenlasten von 500 MW simuliert und weist eine Leitungslänge des Vorlaufs von etwa 330 km bei ca. 4400 angeschlossenen Wärmeübergabestationen sowie mehreren verteilten Wärmeerzeugungsanlagen auf. Die Netze sollen bei Anwendung des Konzeptes wirtschaftlich optimal unter Einhaltung aller nicht-linearen hydraulischen und thermischen Restriktionen betrieben werden können. Somit sollen nicht nur einzelne Aspekte der Netze optimiert werden.

Optimierung

Zum Thema Optimierung von Fernwärmenetzen sei auf die Studien [94], [121] und [107] verwiesen, welche einen Überblick über Methoden für die Optimierung von Fernwärmenetzen geben. Hier soll zunächst auf Projekte eingegangen werden, welche die im vorigen Kapitel beschriebenen *Unit-Commitment*- und *Economic-Dispatch*-Probleme lösen. In Deutschland wurde die bereits vorgestellte Software *BoFiT* in Verbundforschungsvorhaben für die Modellierung und Optimierung von Fernwärmenetzen entwickelt. Diese Software wird auch derzeit noch in Fernwärmenetzen eingesetzt und von der Firma *Volue* vertrieben [135]. Im Rahmen dieser Forschungsvorhaben sollte in *BoFiT* zunächst auch eine Optimierung unter der Verwendung von stationären und instationären Simulationsmodellen erfolgen. Die Modellentwicklungen wurden bereits in Kapitel 3.1 vorgestellt. Optimierungsarbeiten wurden in den Projekten [120], [125] und [25] veröffentlicht. [120] beschreibt ein Pilotprojekt, welches anhand der Fernwärmenetze in Berlin durchgeführt wurde. Die eingesetzten Simulationsmodelle waren allerdings zu langsam für effiziente und einsetzbare Optimierungen. In Hannover und Hamburg wurde ein weiteres Pilotprojekt umgesetzt [125]. Hier wurde zum Beispiel auch die Software *SIR 3S*[®] für die Modellierung des Fernwärmenetzes in Hannover, die Verbesserung einer automatischen Pumpensteuerung und die Online-Visualisierung von Netzdruckdiagrammen verwendet. Zudem wurde eine Optimierung mit *BoFiT* getestet. Die Ergebnisse waren für die Optimierung des Pumpenstroms entweder ähnlich zu realen Fahrweisen oder nicht umsetzbar, da Restriktionen in der Optimierung nicht ausreichend berücksichtigt wurden. Auch ein Soll/Ist-Vergleich der Tageseinsatzplanung konnte

aufgrund zu hoher Rechenzeiten nicht umgesetzt werden. Als Fazit wird beschrieben, dass das System nicht ohne erheblichen Aufwand umgesetzt werden kann. [25] stellt ein weiteres Pilotprojekt der *Hamburgischen Electricitäts-Werke* (HEW) vor mit dem Optimierungsziel der Versorgungssicherheit bei möglichst geringen Betriebskosten. Dafür wurde ein Netzmodell erstellt, welches stationäre und langsame instationäre Vorgänge abbildet. Ein Vergleich mit vorher eingesetzten Modellen zeigte zwar eine hohe Genauigkeit, die Berechnungen waren aber zu langsam für eine Online-Optimierung. Deshalb wurde ein reduziertes Ersatzmodell mit fünf Verbrauchern erstellt, welches zum Zeitpunkt der Veröffentlichung für die Optimierung ausreichte [25, S. 21].

Physikalische Simulationen von Fernwärmenetzen werden heute in *BoFiT* nicht mehr verwendet. Stattdessen besteht *BoFiT* aus zwei Hauptkomponenten. Die Komponente *BoFiT* Prognose [135] erstellt Vorhersagen für Verbrauchslasten von Strom- und Wärmenetzen, für Ertragsprognosen von zum Beispiel regenerativen Erzeugungsanlagen sowie für Preiserwartungen. Anschließend werden diese Prognosen in der Komponente *BoFiT* Optimierung [134] verwendet, um den Erzeugerpark hauptsächlich wirtschaftlich durch das Lösen der *Unit-Commitment*- und *Economic-Dispatch*-Probleme als gemischt ganzzahlige lineare Programme zu optimieren. Für alle Erzeugungsanlagen werden Sollwerte für die Leistungen generiert. Für die Fernwärmenetze werden zudem Vorlauftemperaturen vorgegeben. Hier können Temperaturkurven eingesetzt und mit weiteren Optimierungen kombiniert werden. Das Netz kann so als Speicher verwendet werden, indem zum Beispiel vor Abnahmespitzen zusätzliche Leistung bei höheren Temperaturen in das Netz gespeist wird. Das Netz wird dabei nur als Speicher modelliert, ein tatsächliches Netzmodell wird in *BoFiT* nicht verwendet und ein Großteil der hydraulischen und thermischen Restriktionen daher nicht berücksichtigt.

Gemischt ganzzahlige lineare Optimierungen werden auch in den Projekten [24], [21] und [8] verwendet. [24] löst die Problematik in *MATLAB*[®]. In dem Projekt wird ein Fernwärmenetz mit einer KWK-Anlage, Solarkollektoren, Heizwerken und Speichern berücksichtigt. Das Netz selbst wird aber auch hier nicht simuliert. [21] verwendet die kommerzielle Software *X-press* für die Optimierung eines Fernwärmenetzes mit solarer Einspeisung, verteilten Wärmeerzeugungsanlagen und Speichern. Das Leitungssystem wird nur über einen fixen Wärmeverlustfaktor berücksichtigt. [8] stellt den Status der *TransiEnt*-Bibliothek für die Software *Modelica* vor. *TransiEnt* wurde entwickelt, um gekoppelte Energiesysteme wie zum Beispiel Strom und Wärme zu simulieren. Als Beispiel wird das Versorgungsnetz in Hamburg mit angenommenem Status 2050 zunächst in *MATLAB*[®] linear nach [33] optimiert. Anschließend wird das Netz mit der neu entwickelten Bibliothek simuliert. In den Projekten [3], [38] und [37] wird LP für die Optimierung der Fernwärmenetze verwendet.

[53] setzt eine nicht-lineare Programmierung ein und löst die Einsatzoptimierung somit nicht klassisch wie die bisherigen Projekte und *BoFiT* mit linearen Ansätzen. Im zugrunde liegenden Strom- und Fernwärmenetz werden Windenergieanlagen, Photovoltaik, Heizwerke, Batterien und KWK-Anlagen optimiert. Das Fernwärmenetz selbst wird wiederum nicht simuliert. Ein weiterer *BoFiT*-ähnlicher Ansatz wird in [9] und [137] beschrieben. In den Veröffentlichungen wird das *Balmorel energy dispatch model* vorgestellt und zum Beispiel für die Analyse der Installation von Wärmepumpen im Fernwärmenetz Kopenhagen verwendet. In [111] wird die Software *EnergyPro* angewandt, um einen Ersatz für das Kraftwerk Wedel im Hamburger Fernwärmenetz zu finden.

In den bisher vorgestellten Projekten werden in der Optimierung keine Modelle von Fernwärmenetzen verwendet. In Vorlauftemperaturoptimierungen wird das Netzwerk lediglich als Punktlast simuliert und in einigen Projekten im Anschluss eine Simulation durchgeführt. Nun sollen Projekte vorgestellt werden, welche bereits Simulationsmodelle in der Optimierung verwenden. In [86] wird zusätzlich für die Vorlauftemperaturoptimierung die thermische Dynamik für die Verwendung des Fernwärmenetzes als Speicher berücksichtigt. Dabei werden je nach Betriebspunkt konstante Transportzeiten zu den Wärmeübergabestationen angenommen. Allerdings wird nur eine Wärmeerzeugungsanlage optimiert und die Modellierung ist durch die Annahme konstanter Transportzeiten im Vergleich zu *BoFiT* nur

geringfügig erweitert. Bereits 1995 wurde in dem Projekt [13] ein Simulationsmodell basierend auf der Knotenmethode verwendet, um die Zeitverzögerung der Vorlauftemperatur zu berechnen und anschließend in einer nicht-linearen Optimierung zu berücksichtigen. Hier werden in jedem Schritt neue Simulationen mit dem Knotenmodell durchgeführt, jedoch wird auch in diesem Ansatz nur ein Wärmeerzeugungsstandort, welcher mehrere Anlagen enthalten kann, optimiert.

Komplexere Simulationsmodelle werden in den Arbeiten [133] und [131] verwendet. Das *MATLAB*[®] / *Simulink*[®] Modell basiert auf den Arbeiten [130] und [132]. In den Studien wird eine gemischt ganzzahlige lineare Optimierung der Brennstoffkosten mit einer überlagerten Optimierung mit stationärer Netzsimulation durch einen evolutionären Algorithmus durchgeführt. Netzrestriktionen können somit berücksichtigt werden. Allerdings wird lediglich ein stationäres Modell eines vereinfachten Fernwärmenetzes verwendet. Die Wärmespeicherung im Netz selbst kann nicht berechnet werden.

Instationäre Modelle werden in einigen Arbeiten mit der Software *Modelica* verwendet. In [34] wird mit der bereits erwähnten *Modelica* Bibliothek *TransiEnt* das dynamische Speicherverhalten in Strom- und Wärmenetzen analysiert und eine Linearisierung in eine gemischt ganzzahlige lineare Optimierung übernommen. Die Optimierungsergebnisse werden mit dynamischen Simulationen überprüft. Jedoch wird nur eine 24-Stunden-Optimierung am Vortag berücksichtigt und das Strom- und Fernwärmenetz global betrachtet. Das heißt, die Netzmodelle selbst werden nicht berücksichtigt. Weitere Anwendungsbeispiele für Simulationen mit *Modelica* und anschließender Optimierung sind [106] und [114]. In [106] wird das *Unit-Commitment*-Problem wie in den meisten Ansätzen als gemischt ganzzahliges lineares Programm beschrieben, um den Status verschiedener Wärmeerzeugungsanlagen zu bestimmen. Das *Economic-Dispatch*-Problem wird anschließend als nicht-lineare Optimierung unter der Verwendung von *Modelica* durch den *Interior Point Optimizer (IPOPT)* gelöst. Eine Beispielanwendung auf das Fernwärmenetz in Uppsala (Schweden) zeigt, dass auch ein verhältnismäßig großes Fernwärmenetz mit physikalischen Modellen optimiert werden kann. Allerdings werden in dem stark vereinfachten Fernwärmenetzmodell nur drei Wärmeübergangsstationen berücksichtigt. [114] führt die Arbeiten aus [106] fort und stellt ein auf *Modelica* basierendes Software-Framework vor, welches neben der Optimierung auch Algorithmen zur Aggregation der Netze enthält. Die Vorgehensweise ist nach [114] auf Fernwärmenetze mit 150 bis 200 Wärmeübergangsstationen anwendbar.

Andere Arbeiten beschäftigen sich mit Ansätzen der Pumpenoptimierung sowie Speicherung von Wärmeenergie im Fernwärmenetz, in Speichern oder auch in den angeschlossenen Gebäuden. In [54] wird zunächst ein Teils des Turiner Fernwärmenetzes physikalisch modelliert und durch *proper orthogonal decomposition* (POD) sowie *radial basis functions* (RBF) vereinfacht. Mit beiden Modellen werden Optimierungen durchgeführt, um den Energieverbrauch der Pumpen zu minimieren. Weitere Optimierungsziele werden nicht berücksichtigt. In [55] wird anschließend die virtuelle Speicherung von Wärme in den Gebäuden durch Modifikation ihres thermischen Lastprofils für einen Teil der Turiner Fernwärmenetzes mit 100 Gebäuden untersucht. Eine Optimierung zeigt, dass Spitzenlasten durch den Ansatz reduziert werden können. Ein weiterer Ansatz ist die Erstellung von festen Temperaturkurven für Wärmeerzeugungsanlagen und Wärmeübergabestationen. In [63] werden die primären (zwischen Wärmequelle und Wärmeübergangsstation) und sekundären (zwischen Wärmeübergangsstation und Verbraucher) Temperaturen und Massenströme für verschiedene Außentemperaturen optimiert, um die Pumpen- und Wärmeerzeugungskosten für den jeweiligen Fall zu minimieren. Die nicht-lineare Optimierung wird mit dem Optimierer *fmincon*, einer gradienten-basierten Methode, in *MATLAB*[®] durchgeführt.

Modellprädiktive Regelung

In MPCs werden bereits vorgestellte Ansätze der Optimierung von Fernwärmenetzen um eine Regelung erweitert. Das Prinzip der MPC wurde in Kapitel 2.3 vorgestellt. Einige Forschungsprojekte wie [87] und [104] schlagen dreistufige Optimierungen und Regelungen vor. In [87] besteht die oberste Ebene aus einem *mixed-logical dynamical system*, in welchem binäre und kontinuierliche Entscheidungsvariablen optimiert werden. Auf dieser Ebene werden das *Unit-Commitment*- und *Economic-Dispatch*-Problem hauptsächlich unter Verwendung von Wetter- und Lastprognosen gelöst. In der mittleren Schicht werden die kontinuierliche Dynamik des Systems betrachtet und die Massenströme entsprechend der in der oberen Schicht generierten Leistungs- und Vorlauftemperatursollwerte unter Einhaltung von Restriktionen eingestellt. Dazu gehören zum Beispiel die Leistungsabnahme, die Limitierung der Erzeugereinheiten, Temperaturgrenzen und auch die Dynamik der Wärmespeicherung. Auf der untersten Ebene werden die Prozessvariablen entweder durch schnelle modellprädiktive oder PI(D)-Regelungen umgesetzt. Die vorgeschlagene Strategie wird in Simulationen auf ein Fernwärmenetz in Vrasko (Slowenien) mit 12 km Länge und 187 Wärmeübergabestationen angewandt. [104] wendet eine ähnliche Strategie auf ein kleines Fernwärmenetz in Klæbu (Norwegen) mit einem Wärmeerzeugungsstandort und fünf Verbrauchern an. Hier werden mehrere Wärmeerzeugungsanlagen nicht behandelt und auch das Fernwärmenetz wird auf ein sternförmiges Netz aggregiert.

Für MPCs werden häufig hydraulisch stationäre und thermisch instationäre physikalische Modelle eingesetzt. Beispiele finden sich in [102], [109], [108], [23], [70], [52], [48], [96] und [83], welche im Folgenden genauer beschrieben werden. Für die Optimierung wird eine Reduktion der Wärmeleistungen und somit Kosten angestrebt. [102] unterstreicht das Problem der stark zunehmenden Rechenzeiten bei Problemen steigender Komplexität. Hier wird ein MPC Ansatz auf zwei kleine Wärmenetze angewandt. In einigen Fällen kommt es bereits zu Rechenzeitproblemen oder fehlender Konvergenz der Optimierungsalgorithmen in einzelnen Rechenschritten bei dem komplexeren der beiden Netze. Die Optimierung der Vorlauftemperatur ist in den meisten vorab genannten Projekten enthalten und die verwendeten Modelle werden oft durch verschiedene Methoden vereinfacht. [109] und [108] leiten zum Beispiel ein *MATLAB*[®] Modell basierend auf [110] durch Vorab-Berechnung und lineare Interpolation von Massenströmen und Drücken ab. Dabei wird für einen robusten Einsatz einer MPC vorgeschlagen, immer etwas mehr Leistung in Fernwärmenetze einzuspeisen als von der Regelung vorhergesagt. So sinken zwar mögliche Einsparungen durch die Optimierung, allerdings ist die Versorgungssicherheit in Fernwärmenetzen eine der wichtigsten Restriktionen. Der Anwendungsfall der Projekte ist ein kleines Fernwärmenetz mit zwei Wärmeerzeugungsanlagen und sechs Verbrauchern. In [23] wird *MATLAB*[®] *Simulink*[®] für eine MPC Strategie mit einem 10-Tage-Prädiktionshorizont eingesetzt. Pumpen, Wärmeerzeugungsanlagen, Gebäude und Rohrleitungen werden modelliert und für die Minimierung der eingespeisten Energie angewandt. Jedoch besteht das eigentliche Fernwärmenetz in der Arbeit nur aus zwei Rohrleitungen. In [70] und [52] wird ein in [51] entwickeltes hydraulisch stationäres und thermisch instationäres Modell verwendet. Für kritische Knoten im Netz werden bedingt-parametrische Modelle (*cARX*, *cFIR*) abgeleitet und anschließend in einer *Fuzzy-MPC* verwendet. Bei diesem Ansatz werden die Systemmatrizen der MPC abhängig von *Fuzzy*-Mengen bestimmt. Optimiert werden die operativen Kosten basierend auf Wärmeverlusten und Pumpenkosten des Fernwärmenetzes in Tannheim (Österreich), welches von 84 auf sieben Verbraucher reduziert wird. Die physikalische Netzsimulation wird wie in den vorgestellten Optimierungen häufig in *Modelica* programmiert. In [48] wird eine MPC mit linearisiertem Modell aus dem Projekt [47] erstellt, um die Leistungen der Wärmeerzeugungsanlagen zu optimieren und anschließend Vorlauftemperaturen und Differenzdrücke zu regeln. Dabei wird die Optimierung mit linearisierten Restriktionen und Zielfunktion als gemischt ganzzahliges lineares Programm definiert. Die vorgestellte Methodik ist für große Fernwärmenetze nicht effizient genug, weshalb in dem Projekt auch eine separate *C++* Modellierung umgesetzt wird. Der erstellte MPC ist allerdings nur für Fernwärmenetze mit einer einzelnen Wärmeerzeugungsanlage und nicht für vermaschte Netze geeignet. [96] verwendet einen Ansatz der Vereinfachung, welcher bereits in verschie-

denen Optimierungsprojekten vorgestellt wurde. Hier werden die Zeitverzögerungen der Temperaturausbreitung mit einem Modell in der Software *Apros* berechnet. Anschließend werden basierend auf den Zeitverzögerungen und Wärmelastprognosen die optimalen Vorlauftemperaturen und der Massenstrom einer KWK-Anlage bestimmt. Der vorgeschlagene proaktive Regelungsalgorithmus wird anhand von Simulationen des vereinfachten Fernwärmenetzes in Järvenpää (Finnland) mit 38 km Leitungslänge und 25 zusammengefassten Wärmeübergabestationen getestet und untersucht unter anderem die Zusammenhänge zwischen Wärmeverlusten, Pumpenleistung und minimalem Differenzdruck an den Wärmeübergabestationen bei Anwendung unterschiedlicher Temperaturregelungen.

[45], [46] und [129] untersuchen die Verwendung eines Fernwärmenetzes mit Wärmespeichern, wobei die Modellierung des Netzes selbst größtenteils vernachlässigt wird. Ein Konzept mit Speicherung der Wärmeenergie im Netz wird in [10] entwickelt und auf das Fernwärmenetz in Pongau (Österreich) mit 198 Verbrauchern und 10,3 km Leitungslänge angewandt. Die dafür notwendige Regelung wird vorab mit einem *Modelica* Modell entwickelt. Die Vorlauftemperatur und Differenzdrücke werden anschließend abhängig von der vorhergesagten Wärmeabnahme eingestellt.

Des Weiteren sei die Software *modesto* (*Multi-objective District Heating Energy System Toolbox for Optimization*) erwähnt, welche in *Python* entwickelt wurde, um Fernwärmenetze stationär und instationär sowie weitere Energiesysteme zu modellieren und optimieren. Hier kann ebenfalls eine MPC eingesetzt werden. *modesto* wird in [127] beschrieben und befindet sich zu dem Zeitpunkt der Veröffentlichung (2018) noch in der Entwicklung. Der Fokus eines einfachen Anwendungsbeispiels ist die Berechnung von Zukunftsszenarien über einen Jahresverlauf. Ein ähnliches Beispiel mit *modesto* wird 2019 in [59] vorgestellt. Eine detaillierte Optimierung eines Fernwärmenetzes wird in beiden Veröffentlichungen nicht durchgeführt und weitere Arbeiten wurden seit 2019 nicht veröffentlicht.

Insgesamt wendet keines dieser Projekte eine MPC auf ähnlich komplexe Netze wie die in dieser Arbeit behandelten Fernwärmenetze an und häufig werden nicht alle Aspekte der Optimierung thematisiert. Der Großteil der MPC-Lösungen stellt sich letztendlich als zu komplex heraus und erreicht keine schnellen Optimierungszeiten. Häufig sind starke Vereinfachungen der Modellierung oder der Netzarchitektur erforderlich. Allerdings lassen sich aus den Projekten verschiedene Ansätze ableiten und weiter untersuchen.

Anwendung von Ansätzen des maschinellen Lernens

In den letzten Jahren werden Algorithmen des maschinellen Lernens vermehrt auch für Fernwärmenetze angewandt. Ein Großteil der Projekte beschäftigt sich mit der Vorhersage des Wärmebedarfs, was auch in dem Review [85] festgestellt wird. Die Arbeit hebt außerdem hervor, dass maschinelles Lernen gut für weitere Aufgaben wie die Analyse, Optimierung und Regelung von Fernwärmenetzen geeignet ist und dort noch selten eingesetzt wird. Der Ansatz verspricht kürzere Simulations- und Optimierungsdauern sowie weniger komplexe Modelle. Hauptsächlich werden bisher neuronale Netze eingesetzt, aber auch verschiedene weitere Ansätze verglichen. Im Folgenden werden einige Projekte vorgestellt, welche bereits neuronale Netze oder *Reinforcement Learning* in Optimierungen oder Regelungen einsetzen.

In dem Projekt [77] werden tiefe neuronale Netze verwendet, um die Fahrweise von Fernwärmenetzen basierend auf Messdaten zu erlernen. Hier wird also kein Modell verwendet und neben dem Ansatz, maschinelles Lernen für die Approximation von physikalischen Netzsimulationen zu verwenden, können stattdessen auch Optimierungsergebnisse erlernt werden.

In dem Projekt [140] werden neuronale Netze für den Entwurf von PID-Reglern der Pumpen des Fernwärmenetzes in Afyonkarahisar (Türkei) verwendet. Ziel der Regelung ist die Maximierung der Exergieeffizienz. Die für die Auslegung verwendeten Modelle und Analysen basieren auf den in Kapitel 3.1.2 vorgestellten Projekten [67] und [66]. In [4] wird ein Fernwärmenetz bestehend aus einer

Solarthermieanlage, zwei Gasheizern und zwei Wärmespeichern zunächst in der Software *TRNSYS* modelliert und simuliert. Ein Trainingsdatensatz für ein neuronales Netz wird gebildet, welches das Netzverhalten lernt. Mit dem neuronalen Netz ist anschließend eine effiziente Optimierung des Systems mit bis zu 100 angeschlossenen Gebäuden in *MATLAB*[®] möglich.

Ein ähnliches Ziel verfolgt 2019 das Projekt *Heating Network Navigator (HN-Navi)* [43] [44]. So wird eine optimale Regelung von Fernwärmenetzen basierend auf deterministischen physikalischen Modellen angenähert durch ein neuronales Netz entwickelt. Als Referenz dient eine *Software-in-the-Loop*-Umgebung mit *TRNSYS*. Die Software soll zukünftig an dem Brühl Solarfernwärmenetz in Chemnitz getestet werden. Ergebnisse werden in den Arbeiten nicht präsentiert und wurden seitdem auch nicht veröffentlicht.

Ein *TRNSYS* Modell wird ebenfalls in dem Projekt [65] für das Lastmanagement in Gebäuden verwendet. Das Projekt bezieht sich somit nicht direkt auf die Regelung eines Fernwärmenetzes, behandelt aber eine ähnliche Aufgabenstellung. In dem Projekt soll der Energiebezug für Heiz- und Kühlanwendungen in den Gebäuden durch Verschiebung der Betriebszeiten flexibilisiert werden. Das *TRNSYS* Modell dient dabei als Basis für neuronale Netze, welche in einer MPC für die Erstellung kostenoptimaler Fahrpläne benutzt werden. Die dynamische Optimierungsumgebung wird in *LabVIEW* erstellt.

In [74] werden der Wärmebedarf und die Rücklauftemperatur des Fernwärmenetzes in Kuopio (Finnland) mit neuronalen Netzen basierend auf der Außentemperatur und Prozessdaten vorhergesagt. Die Temperaturentwicklung im Netzwerk wird modelliert. Anschließend wird ein *Brute-Force*-Optimierer eingesetzt, um Vorlauftemperaturen so zu optimieren, dass Pumpkosten und Wärmeverluste unter Berücksichtigung von Temperaturgradienten und somit thermischen Spannungen minimiert werden. Leistungen der Wärmeerzeugungsanlagen werden in diesem Projekt nicht optimiert.

In den Projekten [20] und [27] wird die Optimierung einzelner Wärmeübergabestationen untersucht. [20] verwendet eine MPC basierend auf einem *Recurrent Neural Network* (RNN). Ein Modell einer Wärmeübergabestation mit Wärmespeicher wird entwickelt und die MPC in *LabVIEW* mit einem regelbasierten Regler verglichen. Unter bestimmten Restriktionen ist die Verwendung neuronaler Netze sinnvoll. [27] wendet ebenfalls eine MPC basierend auf neuronalen Netzen an, welche die Abnahme einer Wärmeübergabestation regelt. Die MPC ist in der Lage, den Energieverbrauch der Station verglichen mit einer regelbasierten Regelung zu senken.

Weitere Ansätze, die neuronale Netze verwenden, werden in [136], [30], [50], [64] und [105] vorgestellt. Andere Ansätze wie *Fuzzy Cognitive Maps (FCM)* [81] oder Modellreduktion in *Python* [141] sollen ebenfalls kurz genannt, jedoch nicht ausgeführt werden.

In [142] wird bereits *Deep Reinforcement Learning* angewandt, um die Massenströme in einem Fernwärmenetz zu regeln. Dabei werden sowohl die zentralen Pumpen als auch die Regler von einzelnen Abnehmern angesteuert. Eine Optimierung der Leistungen oder Vorlauftemperaturen wird nicht betrachtet. Weitere Projekte konzentrieren sich auf die Anwendung auf der Gebäude- bzw. Verbraucherseite. [62] verwendet zum Beispiel *Supervised Learning (SL)* für die Lastvorhersage in einem Fernwärmenetz. Anschließend sollen Wärmespeicher in Häusern mittels *Reinforcement Learning* be- oder entladen werden, um Spitzenverbräuche im Fernwärmenetz abzufangen und Rücklauftemperaturen zu reduzieren. Ähnlich wird in [116] vorgegangen und *Reinforcement Learning* eingesetzt, um die Wärmeabnahmeprofile von Gebäuden anzupassen und so Spitzenverbräuche zu verteilen. In [76] wird die Innentemperatur in Gebäuden vorhergesagt und anschließend die optimale Vorlauftemperatur für das Fernwärmenetz geregelt. [89] stellt eine Heizungsregelung mittels Reinforcement Learning vor und vergleicht die Ergebnisse mit einer komplexeren MPC. Die Ergebnisse der komplexeren Simulation sind nur geringfügig besser. In [29] wird ein Gebäude mit Solarkollektoren, einem Wärmerückgewinnungskühler und einer Wärmepumpe in *TRNSYS* modelliert. Das Modell dient als Datenbasis für einen *Reinforcement-Learning*-Algorithmus, der das System optimal regeln soll.

In [88] wird zudem die Anwendung von MPC und *Reinforcement Learning* Regelungen auf Fernwärmenetze untersucht. Beide Ansätze werden in Simulationen des Fernwärmenetzes in Osimo (Italien) mit einem maximalen Wärmebedarf von etwa 9,7 MW mit der derzeitigen regelbasierten- und PID-Regelung der Vorlauftemperaturen und Massenströme verglichen. Wärmeverluste werden dabei durch *Reinforcement Learning* noch deutlicher reduziert als durch die MPC. Zuletzt werden auch die Vor- und Nachteile der beiden Ansätze verglichen. So bestehen die Vorteile einer MPC hauptsächlich in der einfacheren mathematischen Formulierung, dem Finden optimaler Lösungen und der expliziten Berücksichtigung von Restriktionen. Andererseits ist es schwierig, einen Kompromiss zwischen ausreichend genauen und schnellen Netzmodellen zu finden. *Reinforcement Learning* hingegen ist im Einsatz sehr schnell und kann während diesem weiter lernen. Allerdings ist die mathematische Formulierung komplex, optimale Lösungen sind nicht garantiert und Restriktionen werden nicht sicher eingehalten.

4. Regelungskonzept

4.1. Derzeitiger Betrieb der untersuchten Fernwärmenetze

Abbildung 4.1 zeigt das derzeitige Betriebskonzept der in dieser Arbeit behandelten Fernwärmenetze. Zunächst wird eine wirtschaftliche Optimierung durchgeführt. In diese gehen zum Beispiel ökonomische Nebenbedingungen, Wetter- und Lastprognosen sowie Messdaten des tatsächlichen Wärmeverbrauchs ein. Damit wird ein gesamtheitlich wirtschaftlich optimaler Fahrplan für die Strom-, Wärme- und Kälteproduktion für einen bestimmten Zeitraum in der Zukunft berechnet. Die wirtschaftliche Optimierung bestimmt auf der Fernwärmeseite sowohl die Einspeiseleistung als auch Vorlauftemperaturen der unterschiedlichen Wärmeerzeuger und bezieht in den Netzen vorhandene Wärmespeicher mit in die Optimierung ein. Allerdings überprüft sie nur in geringem Maße die hydraulischen Restriktionen der Fernwärmenetze und optimiert Vorlauftemperaturen der Wärmeerzeugungsanlagen basierend auf vereinfachten Speichermodellen.

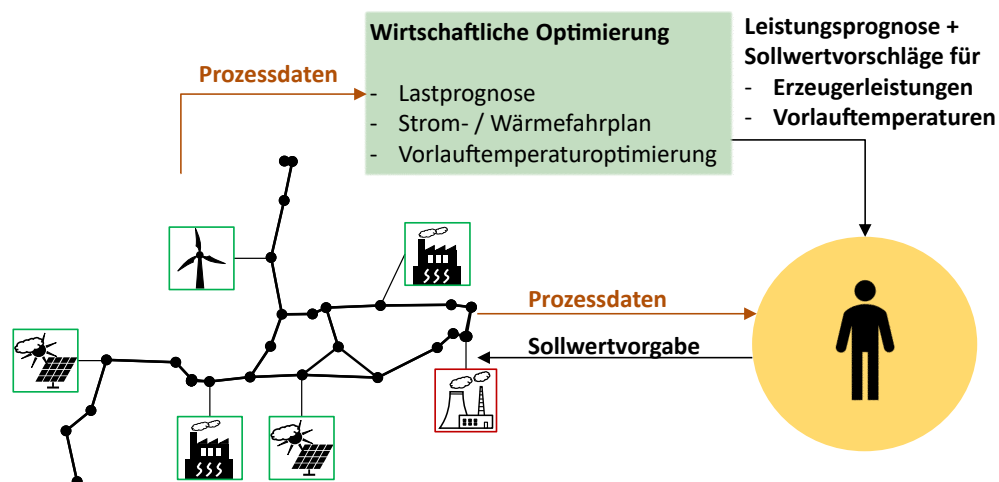


Abbildung 4.1.: Derzeitiges Regelungskonzept für Fernwärmenetze.

Die Sollwerte für Einspeiseleistung und Vorlauftemperatur werden im nächsten Schritt dem Betriebspersonal vorgeschlagen. Das Personal ist für die stabile und möglichst wirtschaftliche Strom- und Wärmeerzeugung verantwortlich. Unter Berücksichtigung der zur Verfügung gestellten Fahrpläne, des aktuellen Betriebszustandes des Erzeugerparcs sowie der Situation am Energiemarkt werden die Anlagen manuell gesteuert. Bei Diskrepanzen zwischen dem wirtschaftlich optimalen vorgeschlagenen Fahrplan und einer realisierbaren Fahrweise muss das Betriebspersonal die Leistungssollwerte der Anlagen anpassen oder neue Anlagen anfahren. Dabei muss auch in den Strom- und Energiehandel eingegriffen und benötigte Ressourcen eventuell nachgekauft oder die Informationen des geänderten Fahrplans an die wirtschaftliche Optimierung bzw. den Energiehandel rückgekoppelt werden. Ebenfalls muss das Personal teilweise in die Druckregelung des Systems eingreifen. Dafür werden alle erforderlichen Prozessdaten des Fernwärmenetzes zur Verfügung gestellt. Auf der Feldebene werden letztendlich die nötigen Differenzdrücke, Mindestdrücke, Durchflüsse und Temperaturen häufig durch PI(D)-Regler weitestgehend automatisiert geregelt. Die genannten Aufgaben sind bereits heute bei einer geringen

Zahl einspeisender Wärmeerzeugungsanlagen herausfordernd. Die Versorgungssicherheit ist dabei jederzeit gewährleistet. Allerdings können aufgrund der Komplexität der Aufgaben und der Netze nicht immer die wirtschaftlich optimalen Sollwerte umgesetzt werden. Wenn die Fernwärmenetze in Zukunft stärker gekoppelt werden und neue, dezentral angelegte regenerative Wärmeerzeugungs- und Speichereinrichtungen hinzukommen, ist davon auszugehen, dass sich diese Problematik verstärken wird. Das bestehende Betriebskonzept soll also wenn möglich verbessert und weiter automatisiert werden.

4.2. Konzept für den optimalen Betrieb von Fernwärmenetzen

Basierend auf den Grundlagen in Kapitel 2, dem Stand der Technik in Kapitel 3 und dem derzeitigen Betriebskonzept in Kapitel 4.1 wird deshalb hier ein neues Regelungskonzept erarbeitet und dessen Umfang eingegrenzt. Das Konzept soll möglichst auf weitere Fernwärmenetze übertragbar sein. Kapitel 3.3 hat Optimierungsansätze und aktuelle Projekte der Optimierung und Regelung vorgestellt. Dabei wurden auch die *Unit-Commitment*- und *Economic-Dispatch*-Probleme erläutert, welche Wärmeerzeugungsanlagen für die Produktion bestimmen und anschließend die Leistungswerte zwischen diesen Anlagen verteilen. Eine Lösung der Probleme ist im vorliegenden Fall durch die wirtschaftliche Optimierung gegeben. Allerdings werden nicht-lineare hydraulische und thermische Restriktionen nicht berücksichtigt. In Kapitel 3.3.1 wurde beschrieben, dass die Lösung der *Unit-Commitment*- und *Economic-Dispatch*-Probleme unter Verwendung nicht-linearer Zielfunktionen und Nebenbedingungen derzeit nicht effizient möglich ist. Stattdessen wurde eine Trennung in die lineare Lösung des *Unit-Commitment*- und nicht-lineare Lösung des *Economic-Dispatch*-Problems vorgeschlagen. Hier soll ein ähnliches Konzept untersucht werden. Da wirtschaftliche Optimierungen bei vielen Netzbetreibern bereits im Einsatz sind und zum Beispiel gekoppelte Strom-, Fernwärme- und Fernkältenetze optimieren, soll die wirtschaftliche Gesamtoptimierung weiterverwendet werden. Diese dient anschließend als Startpunkt für eine MPC, welche für die Einhaltung aller hydraulischen und thermischen Restriktionen sorgt und ein Fernwärmenetz automatisiert regeln soll. Das vorgeschlagene Regelungskonzept ist in Abbildung 4.2 dargestellt.

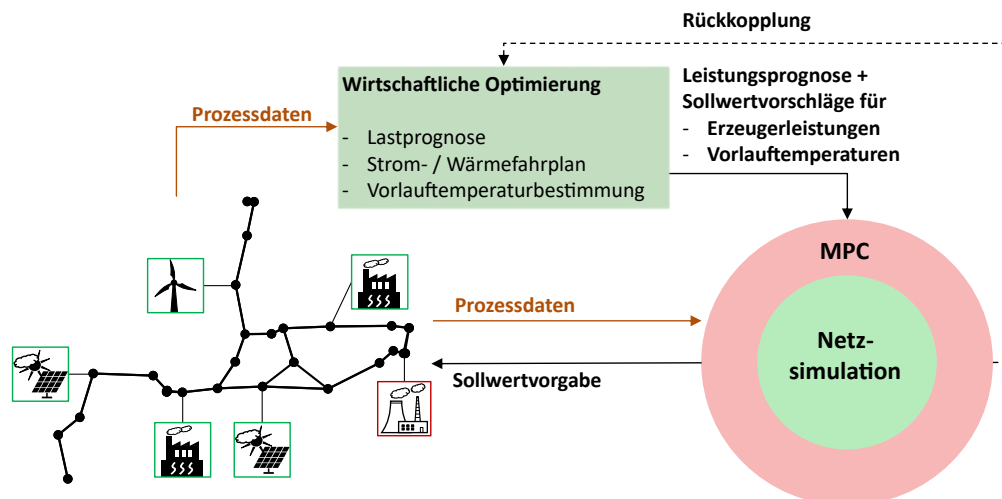


Abbildung 4.2.: Konzept für ein automatisiertes Fernwärmenetz unter Verwendung einer MPC.

Im ersten Schritt bleibt also die wirtschaftliche Optimierung durch Lösung des *Unit-Commitment*- und des *Economic-Dispatch*-Problems bestehen. Anschließend werden optimierte Einspeiseleistungs- und Vorlauftemperatursollwerte für die ausgewählten Wärmeerzeugungsanlagen an eine Schicht mit MPC gegeben. Ebenso werden prognostizierte und aus der Feldebene gemessene Wärmeverbräuche übermittelt. Basierend auf diesen Werten wird eine nicht-lineare physikalische Netzsimulation durchgeführt,

welche die hydraulischen und thermischen Zustände des Netzes berechnet. Kann das Netz nach wirtschaftlich optimaler Vorgabe nicht ausreichend versorgt werden, müssen die Einspeiseleistungs- und Vorlauftemperatursollwerte verändert werden. Die Optimierung der MPC muss hierzu umplanen. Anschließend werden physikalisch umsetzbare Sollwerte an die Feldebene des Fernwärmenetzes übergeben und zur wirtschaftlichen Optimierung bzw. zum Energiehandel rückgekoppelt. Im letzten Schritt bleiben die vorhandenen Regelungen auf der Feldebene bestehen.

Im Rahmen des Projekts wurde entschieden, nicht direkt eine MPC Lösung zu entwickeln, da das Netz weiter von Betriebspersonal gesteuert werden soll. Somit ändert sich das Vorgehen zu dem in Abbildung 4.3. Statt einer MPC wird eine nicht-lineare Optimierung mit Netzsimulation durchgeführt und die Sollwerte an das Personal weitergegeben. Aufgrund der Komplexität des Problems soll bei einer Inbetriebnahme schrittweise vorgegangen werden. Der Großteil der Sollwerte sollte von dem Personal gesetzt werden. An einigen Stellen ist eine direkte Aufschaltung durch eine MPC möglicherweise schon sinnvoll. Nach ausreichender Erprobung könnte eine MPC selbst die Regelung der Netze übernehmen und vom Betriebspersonal überwacht werden. Die Optimierung müsste dann um eine Rückkopplung der Messwerte und eine direkte Aufschaltung der Sollwerte auf das Netz ergänzt werden, kann aber voraussichtlich in ähnlicher Form bestehen bleiben. Da zunächst keine MPC entwickelt werden soll, wird in dieser Arbeit im Detail nur eine Optimierung der Sollwerte untersucht.

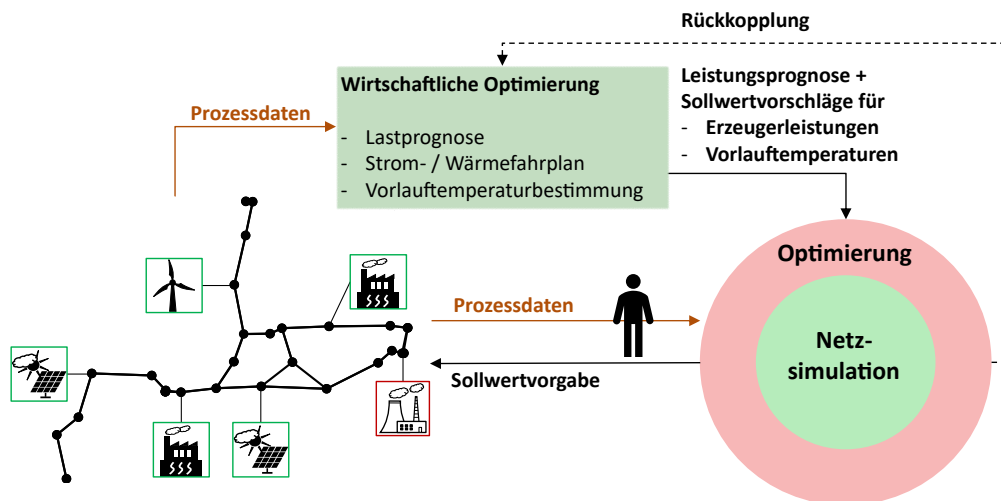


Abbildung 4.3.: Konzept für ein halb-automatisiertes Fernwärmenetz unter Verwendung einer nicht-linearen Optimierung.

Das vorgestellte Konzept ändert insgesamt nur wenige Aspekte der bestehenden Betriebsweise. Wirtschaftliche Optimierungen, Betriebspersonal und Regelungen auf der Feldebene werden nicht verändert. Es wird lediglich eine weitere Automatisierungsschicht zwischen wirtschaftlicher Optimierung und Betriebspersonal hinzugefügt, welche Einspeiseleistungs- sowie Vorlauftemperatursollwerte wenn nötig umplanen kann. Zu beachten ist, dass bei Umplanung und Abweichung von den wirtschaftlich optimierten Sollwerten eine Rückkopplung zum Energiehandel erfolgen muss. Diese kann entweder von der Optimierungsschicht oder wie bisher vom Betriebspersonal übernommen werden.

Eine Alternative zu dem vorgestellten Konzept wäre eine Regelung durch *Reinforcement Learnings*. Der Ansatz wurde in Kapitel 3.3.2 mit einer MPC verglichen und soll hier nicht zum Einsatz kommen, da die Einhaltung von Restriktionen nicht garantiert werden kann. *Reinforcement Learning* stellt aber eine interessante Alternative zu einer MPC dar und kann in weitergehenden Forschungsvorhaben untersucht werden.

4.2.1. Anforderungen an die Optimierung und Netzsimulation

In diesem Kapitel werden Anforderungen an die Optimierung und Modellierung basierend auf den Informationen in Kapiteln 3.1 und 3.3 bestimmt. Es soll eine Optimierung entwickelt werden, welche die Sollwerte der wirtschaftlichen Optimierung möglichst genau einhält. Die Sollwerte werden also nur so weit optimiert, dass alle hydraulischen und thermischen Restriktionen eingehalten werden. Dafür kann vorab eine Priorisierung der Wärmeerzeugungsanlagen und deren Vorlauftemperaturen erfolgen, um festzulegen, wie wenn nötig umgeplant wird. Hier könnten auch weitere physikalische oder wirtschaftliche Optimierungskriterien festgelegt werden. Die Vorlauftemperatur soll, wenn möglich, in diesem Ansatz nicht verändert und wie von der wirtschaftlichen Optimierung vorgeschlagen verwendet werden. Der Schwerpunkt liegt also auf der Einhaltung der physikalischen Restriktionen. Dazu zählen die Limitierungen der Wärmeerzeugungsanlagen und Wärmespeicher, welche bei der Umplanung nicht überschritten werden dürfen, wie maximale Einspeiseleistungen, Vorlauftemperaturen und Pumpengrenzen. Weiterhin müssen hydraulische Restriktionen wie die Limitierung von Strömungsgeschwindigkeiten und Drücken sowie vorgeschriebene Differenzdrücke an allen Wärmeübergabestationen eingehalten werden. Zusätzlich werden thermische Bedingungen wie Grenzkurven und Gradienten sowie minimale Vorlauftemperaturen an den Wärmeverbrauchern vorgegeben. Bei Bedarf soll es möglich sein, der Optimierung weitere, aus der Netzsimulation generierte Restriktionen zu übergeben. In Netzen, in welchen das Betriebspersonal Pumpengruppen für die Differenzdruckregelung manuell auswählen muss, soll diese Auswahl möglichst durch die Optimierung erfolgen. Auch Druckerhöhungsanlagen werden, wo möglich, mit optimiert. Allerdings soll keine detaillierte hydraulische Reglerauslegung stattfinden. Durch die Auswahl von unterschiedlichen Betriebsmodi entstehen voraussichtlich Unstetigkeiten in den Zielfunktionen oder Restriktionen. An dieser Stelle muss untersucht werden, welche Optimierungsalgorithmen sich für diese Probleme eignen und ob eine effiziente Umsetzung möglich ist.

Gegenstand weiterer Analysen soll eine Optimierung sein, welche zusätzlich versucht, die Wärme- und Pumpenleistungen zu minimieren. Die Optimierung soll dabei die Einspeiseleistungen möglichst einhalten, welche auch einen Einfluss auf das Strom- oder Fernkältenetz haben. Andere Einspeiseleistungs- und Vorlauftemperatursollwerte können, wenn möglich, reduziert werden. Durch diesen Ansatz werden Grenzen der Fernwärmeversorgung mit gegebenem Sicherheitsabstand ausgereizt. Einige der Restriktionen wie Grenzkurven und Gradienten für die Vorlauftemperaturen werden bisher hauptsächlich heuristisch oder basierend auf Messdaten festgelegt. Durch Analysen und genaue Abbildung der Fernwärmenetze in einer Simulation scheint eine Anpassung dieser Grenzwerte möglich und soll untersucht werden. In diesem zweiten Optimierungsansatz sollen die in Kapitel 3.3.1 und 3.3.2 vorgestellten Ansätze der Vorlauftemperaturoptimierung und Speichernutzung untersucht werden.

An die Modellierung wird neben ausreichender Genauigkeit die Anforderung der kurzen Rechenzeit gestellt. Die wirtschaftliche Optimierung liefert alle 15 Minuten neue Sollwerte, welche in der Zwischenzeit möglichst schnell bezüglich der Restriktionen optimiert werden sollen. Bei der Weitergabe der Sollwerte an das Betriebspersonal ist eine relativ geringe Frequenz und die Vorgabe robuster Sollwerte sinnvoll. Bei Anwendung einer MPC sollte die Berechnungsfrequenz neuer Sollwerte möglichst hoch liegen, um eine Abweichung zwischen tatsächlicher Fahrweise und Regelung zu vermeiden.

Es wird davon ausgegangen, dass eine stationäre hydraulische Modellierung der Netze ausreicht. Instationäre hydraulische Berechnungen steigern die Simulationszeiten und sind nur für detaillierte Reglerauslegungen der Hydraulik notwendig. Für die thermische Simulation sollen anfangs ebenfalls stationäre Simulationen eingesetzt werden, um Rechenzeiten möglichst gering zu halten. Anschließend sollen die Ergebnisse mit instationären thermischen Simulationen verifiziert werden. Es wird erwartet, dass bei relativ konstanten Vorlauftemperaturen oder bei Netzen mit kurzen Ausbreitungswegen thermisch stationäre Simulationen ausreichen. Dies könnte folglich auch für große Fernwärmenetze

mit dezentral verteilter Wärmeeinspeisung gelten. Bei nicht ausreichenden Optimierungen durch stationäre Simulationen muss eine thermisch instationäre Optimierung in Betracht gezogen werden. Für den zweiten Optimierungsansatz, in welchem auch Vorlauftemperaturen so weit möglich abgesenkt werden, wird davon ausgegangen, dass instationäre Simulationen notwendig sind, da durch dynamische Änderungen der Vorlauftemperatur auch der Massenstrom und die Strömungsgeschwindigkeiten beeinflusst werden. Ohnehin kann eine Optimierung der Nutzung der Fernwärmenetze als Speicher nur mit instationären thermischen Berechnungen erfolgen. Bei stationären Simulationen breiten sich Temperaturen im Netz sofort aus. Die Massenströme werden in der Simulation entsprechend so eingestellt, dass alle Wärmeabnahmen gedeckt werden. Das System befindet sich im Gleichgewicht. Bei instationären Simulationen kann die Vorlauftemperatur an Wärmeerzeugungsstandorten erhöht werden, während der Massenstrom noch von den an Wärmeübergangsstationen anliegenden, niedrigeren Temperaturen bestimmt wird. Somit wird bei gleichem Massenstrom und höherer Vorlauftemperatur eine höhere Leistung in das Netz gefahren und gespeichert.

4.2.2. Vorgehensweise

Entsprechend dem vorgestellten Konzept und den Anforderungen wird in diesem Kapitel die weitere Vorgehensweise beschrieben. Zunächst muss ein geeignetes Simulationsmodell der Fernwärmenetze entwickelt werden. Nach Kapitel 3.1 kommen hauptsächlich Knoten-Kanten-Modelle in Frage. In Kapitel 5 werden die Modellbildung von Knoten-Kanten-Modellen dargelegt und erarbeitete Modelle beschrieben. Diese basieren zunächst auf bereits vorhandenen Modellen. Weitere Ansätze und auch kommerzielle Umsetzungen werden untersucht. Modelle der Software *STANET*[®] und *SIR 3S*[®] werden für die behandelten Fernwärmenetze bereits genutzt. Zudem kommt die Annäherung von Modellen durch Algorithmen der Systemidentifikation oder des maschinellen Lernens infrage. Es sollen dabei nur hydraulisch stationäre Modelle zum Einsatz kommen. Für die thermische Berechnung werden sowohl stationäre als auch instationäre Modelle entwickelt.

Anschließend werden die Modelle hinsichtlich ihrer Anwendbarkeit in einer Optimierung getestet. Die Optimierung wird zunächst so umgesetzt, dass wirtschaftliche Sollwerte möglichst direkt aufgebracht und Leistungen und Vorlauftemperaturen nicht oder wenig geändert werden. Anschließend wird eine weitere Optimierung zum Beispiel durch Absenkung der Vorlauftemperaturen untersucht. Lokale und globale Optimierungsalgorithmen entsprechend Kapitel 2.2 sollen getestet werden.

Die beschriebenen Ansätze werden auf vier Fernwärmenetze mit zunehmender Komplexität angewandt, für welche für die vorliegende Arbeit Daten zur Verfügung stehen. Zuerst ist die Untersuchung eines Netzes mit zentraler Wärmeeinspeisung sinnvoll. Als Beispiel wird ein Fernwärmenetz herangezogen, welches schon in der Arbeit von Pierl [100] untersucht wurde. Für das Netz mit nur einem Wärmeerzeugungsstandort ist ein guter Modellierungs- und Datenstand vorhanden. An dem Standort kommen die Wärmeströme von zwei Wärmeerzeugungsanlagen zusammen und können mit zwei Wärmespeichern das Netz flexibel versorgen. Neben der Verteilung der Erzeugung auf die vier Einheiten kann auch der Einfluss der Vorlauftemperatur analysiert werden. Für Netz 1 werden detaillierte Simulationsmodelle erarbeitet und stationäre sowie instationäre Optimierungen getestet.

Die Erkenntnisse sollen anschließend auf drei Netze angewandt werden, in welche Wärme jeweils an mehr als einem Standort und teilweise aus regenerativen Energien sowie Speicheranlagen eingespeist wird. Hier kann also die Optimierung der verteilten Wärmeeinspeisung untersucht werden. Als regenerative Wärmequelle kommt dabei in Netz 3 die Geothermie zum Einsatz, durch welche zusätzliche Restriktionen für die Optimierung notwendig werden. In den Netzen sind des Weiteren Druckerhöhungsanlagen vorhanden, welche in eine Optimierung integriert werden können.

Weitere regenerative Wärmeerzeugungsanlagen werden bisher nicht in die Optimierung aufgenommen. Zum Beispiel wird die direkte Einspeisung aus Solaranlagen, welche die zusätzliche Herausforderung

einer stark wetterabhängigen Leistung stellen, nicht untersucht. Zudem werden die Fernwärmenetze in dieser Arbeit nur im vollständigen Zustand betrachtet. Das heißt, Ausnahmefälle wie Gebietsabsperungen oder mögliche Erweiterungen der Netze durch hinzukommende Netzgebiete oder Wärmeerzeugungsanlagen werden nicht berücksichtigt.

4.2.3. Limitierungen des Konzepts

Einige Limitierungen des vorgestellten Konzepts können bereits aufgezeigt werden. Das Fernwärmenetz wird hier zweistufig optimiert. Im ersten Schritt findet eine lineare wirtschaftliche Optimierung statt, welche das gesamte Strom-, Fernwärme- und Fernkältenetz optimiert. Im zweiten Schritt können auch nicht-lineare Restriktionen berücksichtigt werden und ein komplexeres Modell des Fernwärmenetzes abbilden. So soll die Einhaltung aller Restriktionen sichergestellt und das Netz automatisiert nah am wirtschaftlichen Optimum gefahren werden. Durch die Umplanung bei Nichteinhaltung physikalischer Restriktionen wird das wirtschaftliche Optimum gegebenenfalls nicht genau erreicht. Auch durch die Weitergabe der Sollwerte an das Betriebspersonal entstehen Abweichungen und es wird keine tatsächliche Regelung umgesetzt. Die von der physikalischen Optimierung generierten Sollwerte müssen möglichst robust sein, damit bei Abweichungen die Versorgungssicherheit nicht gefährdet ist. Es wird hier allerdings davon ausgegangen, dass die Erzeugung von Sollwerten mit Einhaltung aller Restriktionen robuster ist als die alleinige Vorgabe durch wirtschaftliche Optimierungen.

Zu beachten ist auch, dass nicht alle Eingangsdaten der Optimierung genau bekannt sind. Zum Beispiel weicht die Wärmebedarfsprognose im ersten Optimierungsschritt in der Regel von der Realität ab. Auch die aus der Optimierung resultierenden wirtschaftlich optimalen Einspeiseleistungen werden somit vom tatsächlichen Optimum abweichen. Nach Kapitel 2.2 müsste eine stochastische Optimierung angewandt werden, welche unsichere Eingangsdaten berücksichtigt. Die Optimierung soll hier aber zunächst nur deterministisch durchgeführt werden. Eine Optimierung unter Berücksichtigung der Unsicherheiten könnte anschließend durch mehrere deterministische Optimierungen im Bereich einer festzulegenden Unsicherheit der Variablen erfolgen. Eine Demonstration der Funktionsfähigkeit des vorgeschlagenen Konzepts kann somit bereits durch eine einzelne deterministische Optimierung erfolgen. Die Berücksichtigung von Unsicherheiten würde die Rechenzeiten weiter erhöhen.

Die Optimierung selbst wird im Wesentlichen durch Genauigkeit und Rechenzeit der Modellierung sowie die Anzahl der Variablen und Nebenbedingungen (Restriktionen) limitiert. Die Anzahl der Variablen und Nebenbedingungen kann dabei nicht beeinflusst werden. Der Detaillierungsgrad der Modellierung hingegen schon. Je genauer die Modellierung, desto bessere Sollwerte können ermittelt werden. So muss untersucht werden, welcher Detaillierungsgrad der Modellierung ausreicht und in Optimierungen mit limitierter Rechenzeit einsetzbar ist. Entsprechend müssen resultierende Fehler der Modellrechnungen berücksichtigt werden.

Eine weitere Limitierung des Konzepts ist die Umsetzung der Bestimmung von Pumpengruppen für die Druckregelung oder Betriebsarten für Druckerhöhungsanlagen. Die entstehenden unstetigen Funktionen mit wechselnden Betriebsarten entsprechen einer gemischt ganzzahligen nicht-linearen Optimierung, welche nur unter erheblichem Rechenaufwand umsetzbar ist. Eine geeignete Lösung wie zum Beispiel eine weitere Trennung der Optimierungsschritte muss gefunden werden.

5. Thermo-hydraulische Modellierung von Fernwärmenetzen

Basis für das in Kapitel 4.2 vorgestellte Regelungs- und Optimierungskonzept ist ein Simulationsmodell, welches die thermo-hydraulischen Zustände Druck, Durchfluss und Temperatur an den relevanten Orten im Netz berechnen und vorhersagen kann. In den folgenden Kapiteln werden die dafür relevanten physikalischen Gleichungen der in Kapitel 2.1 vorgestellten Komponenten erläutert. Die hydraulische Modellierung wird dabei ähnlich eines vorhandenen Modells von Pierl [100] durchgeführt. Ein Großteil der Theorie wird auch aus den Arbeiten von Icking [61] und Löwen [79] verwendet. In Kapitel 5.1 wird zunächst die stationäre hydraulische Berechnung beschrieben. In den Kapiteln 5.2 und 5.3 folgen die stationären und instationären thermischen Gleichungen. Die Theorie kann anschließend mittels verschiedener Modellierungsansätze umgesetzt werden. Nach Kapitel 3.1 eignen sich Knoten-Kanten-Modelle gut für die Programmierung, aber auch Programme mit grafischen Schnittstellen wie *Simulink*[®] für *MATLAB*[®] oder *Dymola* für *Modelica* können verwendet werden. In Kapitel 5.4 wird die Modellierung mit *Simulink*[®] vorgestellt und anschließend in Kapitel 5.5 Knoten-Kanten-Modelle. Hier kommen auch die bereits verwendeten Programme *STANET*[®] und *SIR 3S*[®] infrage.

Die modellierten Fernwärmenetze werden anschließend zusammen mit der Modellauswertung in Kapitel 6 vorgestellt. Dabei wird im Detail auf Ansätze der Systemidentifikation und des maschinellen Lernens eingegangen.

5.1. Stationäre hydraulische Berechnung

Im Folgenden werden die relevanten Gleichungen für die hydraulische Berechnung von Fernwärmenetzen vorgestellt, welche auf einer Vereinfachung der *Navier-Stokes*-Gleichungen unter Berücksichtigung der Kontinuitätsgleichungen für Massen-, Impuls- und Energieerhaltung basieren. Die Herleitung wird nicht beschrieben und kann zum Beispiel in der Arbeit von Pierl [100] nachgelesen werden. Die hier vorgestellten Gleichungen werden hauptsächlich entsprechend der Arbeit von Icking [61] verwendet. Dabei werden reale Fluide unter Berücksichtigung von Reibung berücksichtigt. Die Stoffgrößen Dichte, Viskosität und Wärmekapazität sind somit druck- und temperaturabhängig. Die Stoffgrößen können nach dem Standard *IAPWS IF-97 (International Association for Properties of Water and Steam Industrial Formulation 1997)* berechnet werden. Die Programmierung des Standards kann theoretisch selbst umgesetzt oder auf verfügbare Implementationen zurückgegriffen werden. In *MATLAB*[®] ist zum Beispiel die Funktion *X Steam* von Magnus Holmgren [60] frei verfügbar. Alternativ können auch Tabellen der Fluideigenschaften erstellt und während der Berechnung für die jeweiligen Drücke und Temperaturen interpoliert werden. Diese Vorgehensweise beschleunigt die Berechnung der Stoffwerte.

Hydraulische Modellierung von Rohrleitungen

Die Berechnung des Druckverlusts Δp über eine gerade Rohrleitung kann mittels der *Darcy-Weisbach-Gleichung* (5.1) erfolgen. Die grundlegende Formel ist in [115, S. 139] beschrieben und die massenstromabhängige Darstellung beispielsweise in [100, S. 120].

$$\Delta p_r = \lambda \frac{8l}{\pi^2 d^5} \frac{1}{\rho} |\dot{m}| \dot{m} \quad (5.1)$$

λ steht dabei für den Rohrreibungsbeiwert, l und d für die Länge und den Durchmesser der Leitung und ρ für die Dichte des Fluids. Da in Fernwärmenetzen Rohranfang und -ende nicht unbedingt auf gleicher Höhe liegen, wird noch ein Druckverlust Δp durch den Höhenunterschied Δh nach Gleichung (5.2) hinzugefügt [61, S. 20]:

$$\Delta p_h = -\rho \cdot g \cdot \Delta h \quad (5.2)$$

$$\text{mit } \Delta h = h_{\text{anfang}} - h_{\text{ende}} \quad (5.3)$$

Durch Krümmungen, Verengungen, Einbauten oder andere Zusatzwiderstände der Rohre können weitere Druckverluste entstehen, welche mittels des zusätzlichen Widerstandsbeiwertes ζ nach Gleichung (5.4) berechnet werden [17, S. 180] [61, S. 20].

$$\Delta p_{\text{zus}} = \zeta \cdot \frac{8}{\pi^2 \cdot d^4} \cdot \frac{1}{\rho} \cdot |\dot{m}| \cdot \dot{m} \quad (5.4)$$

Der gesamte Druckverluste ergibt sich aus Gleichung (5.5):

$$\Delta p = \frac{8}{\pi^2 \cdot d^4} \cdot \frac{1}{\rho} \cdot \left(\frac{l}{d} \cdot \lambda + \zeta \right) \cdot |\dot{m}| \cdot \dot{m} - \rho \cdot g \cdot \Delta h \quad (5.5)$$

Die Gleichung kann durch die Einführung des Rohrleitungswiderstands R in Gleichung (5.6), auch Rohrkonstante genannt, weiter vereinfacht werden und resultiert in dem Zusammenhang (5.7) zwischen Differenzdruck und Massenstrom [61, S. 20] :

$$R(\dot{m}) = \frac{8}{\pi^2 \cdot d^4} \cdot \frac{1}{\rho} \cdot \left(\frac{l}{d} \cdot \lambda + \zeta \right) \cdot |\dot{m}| \quad (5.6)$$

$$\Delta p = R(\dot{m}) \cdot \dot{m} - \rho \cdot g \cdot \Delta h \quad (5.7)$$

Der Rohrreibungsbeiwert λ wird für laminare und turbulente Strömungen unterschiedlich bestimmt und hängt von der Reynoldszahl Re ab, welche durch Gleichung (5.8) definiert ist [17, S. 151].

$$Re = \frac{v \cdot d \cdot \rho}{\mu} \quad (5.8)$$

v ist dabei die Strömungsgeschwindigkeit und μ die Viskosität. Laminare und turbulente Strömungen können durch Grenzen der Reynoldszahl definiert werden. Für eine Reynoldszahl kleiner oder gleich der kritischen Reynoldszahl $Re \leq 2320$ ist die Strömung laminar und der Rohrreibungsbeiwert lässt sich wie folgt berechnen [17, S. 181]:

$$\lambda = \frac{64}{Re} \quad (5.9)$$

Bei einer turbulenten Strömung können je nach Rohrrauigkeit unterschiedliche Approximationen des Rohrreibungsbeiwertes verwendet werden. Je nach Rohrrauigkeit k kann dabei ein Rohr als hydraulisch

glatt, hydraulisch rau oder im Übergangsbereich liegend definiert werden. Hier wird angenommen, dass die Rohre im Übergangsbereich liegen und dementsprechend die Näherung nach *Colebrook-White* [17, S. 182] angewandt werden kann:

$$\lambda = \left[2 \cdot \log \left(\frac{2.51}{Re \cdot \sqrt{\lambda}} + \frac{0.27 \cdot k}{d} \right) \right]^{-2} \quad (5.10)$$

λ muss dabei iterativ berechnet werden. Der Rohrreibungsbeiwert kann auch durch andere Approximationen, welche die Berechnung teilweise noch weiter vereinfachen, bestimmt werden und somit unterschiedliche Ergebnisse bei dem Vergleich mehrerer Modelle verursachen.

Hydraulische Modellierung von Ventilen

Ventile verhindern im geschlossenen Zustand den Durchfluss vollständig und sperren somit zwei Leitungen voneinander ab. Diese abgesperrten Verbindungen werden dann in Berechnungen nicht mehr berücksichtigt. Bei vollständiger Öffnung kann ein Ventil näherungsweise als Leitung betrachtet werden, gegebenenfalls mit einem zusätzlichen spezifischen Widerstand wie in Gleichung (5.4). Beim Einsatz in der Druck- oder Massenstromregelung kann der Druckverlust eines Ventils abhängig vom Massenstrom \dot{m} und Widerstandsbeiwert ζ_{Vent} je nach Öffnungswinkel φ berechnet werden [61, S. 23]:

$$\Delta p = |\dot{m}| \cdot \dot{m} \cdot \frac{1}{\rho \cdot d^4} \cdot \zeta_{Vent}(\varphi) \quad (5.11)$$

Der Zusammenhang zwischen Öffnungswinkel und Widerstandsbeiwert eines Ventils wird Kennlinien entnommen.

Hydraulische Modellierung von Pumpen

Die Pumpen führen dem Strömungsmedium Energie zu und erhöhen dessen Druck. Dabei gilt die Massenerhaltung, der Massenstrom vor und hinter der Pumpe bleibt gleich. Die Pumpenkennlinie beschreibt den Zusammenhang zwischen Durchfluss und Druckerhöhung. Abbildung 5.1 zeigt beispielhaft eine Pumpenkennlinie. Bei einem bestimmten Volumenstrom kann die Pumpe den Druck des Mediums um einen aus der Pumpenkennlinie entnehmbaren Differenzdruck erhöhen. Der Volumenstrom wird oft in $\text{m}^3 \text{h}^{-1}$ angegeben, der Differenzdruck in bar oder auch als Förderhöhe in m. In diesem Fall entspricht der Druck einer Wassersäule in Metern.

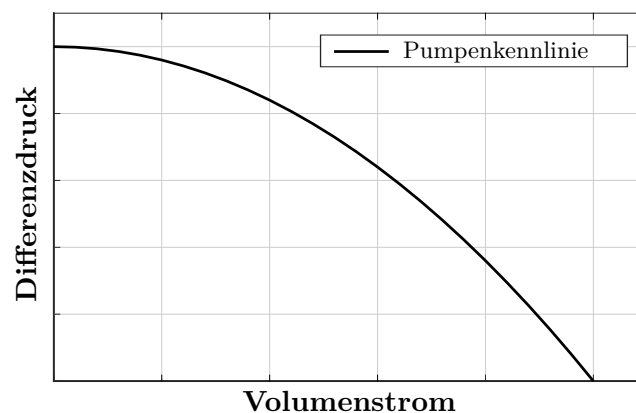


Abbildung 5.1.: Der Zusammenhang zwischen Druckdifferenz und Volumenstrom einer Pumpe dargestellt in der Pumpenkennlinie.

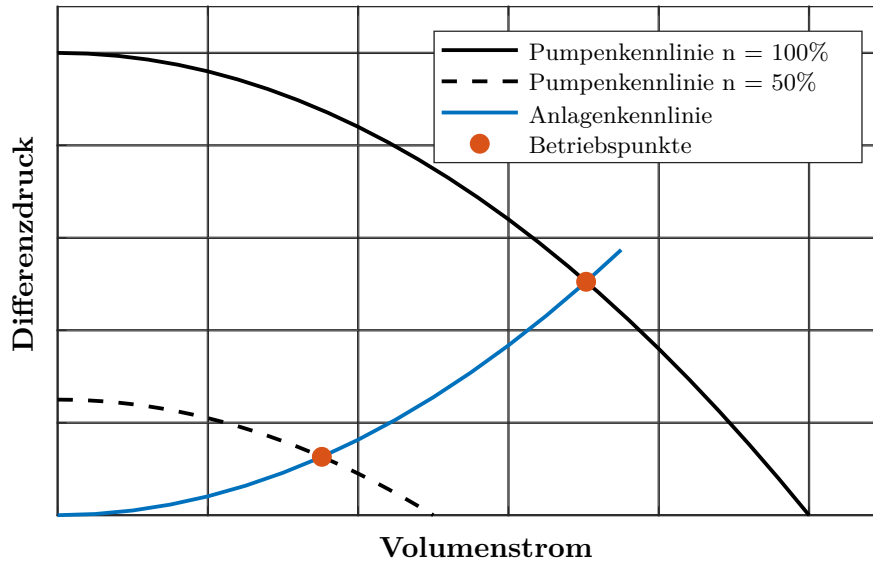


Abbildung 5.2.: Beispiel einer Pumpenkennlinie bei 50 % und 100 % der Drehzahl zusammen mit der Anlagenkennlinie.

Die Pumpenkennlinie wird jeweils bei einer bestimmten Drehzahl angegeben. Für niedrigere oder höhere Drehzahlen ergibt sich bei einem gegebenen Volumenstrom entsprechend ein niedrigerer oder höherer Differenzdruck. Der geförderte Volumenstrom \dot{V} ist dabei proportional zur Drehzahl n , der Differenzdruck Δp proportional zum Quadrat der Drehzahl [92, S. 135]:

$$\frac{\dot{V}_1}{\dot{V}_2} = \frac{n_1}{n_2} \quad (5.12)$$

$$\frac{\Delta p_1}{\Delta p_2} = \left(\frac{n_1}{n_2}\right)^2 \quad (5.13)$$

Abbildung 5.2 zeigt beispielhaft die Kennlinie einer Pumpe bei voller und halber Drehzahl. Der starke Abfall des Volumenstroms und Differenzdruckes ist ersichtlich. Zusätzlich ist in dieser Abbildung die Anlagenkennlinie dargestellt. Sie beschreibt das Verhalten der Druckverluste im Netz, welche mit steigendem Volumenstrom ebenfalls ansteigen. Die Punkte, an welchen die Anlagenkennlinie die Pumpenkennlinien schneidet, sind jeweils die Betriebspunkte der Anlage. An diesen Punkten kann die Pumpe den Druckverlust im Netz ausgleichen. Die Pumpenkennlinie kann als Polynom zweiter Ordnung abhängig vom Massenstrom beschrieben werden [61, S. 21] [112]:

$$\Delta p = \sum_{i=0}^2 a_i(n) \cdot \dot{m}^i \quad (5.14)$$

Die Koeffizienten a_i des Polynoms sind dabei abhängig von der Drehzahl und können folgendermaßen basierend auf bekannten Koeffizienten bei der Drehzahl n_0 beschrieben werden [61, S. 21] [112]:

$$a_i = a_i(n_0) \cdot \left(\frac{n}{n_0}\right)^{2-i} \quad (5.15)$$

Bei einer Regelung im Modell muss also der Massenstrom an einer Pumpe zum Beispiel so eingestellt werden, dass ein minimaler Differenzdruck an den Netzschlechtpunkten gehalten wird. Aus dem daraus resultierenden Differenzdruck, welchen die Pumpe überwinden muss, kann die notwendige Pumpen-

drehzahl errechnet werden.

Die benötigte Pumpleistung P_p , lässt sich mittels der Förderhöhe h , dem Volumenstrom \dot{V}_p und dem Wirkungsgrad η_p der Pumpe berechnen [92, S. 136]:

$$P_p = \frac{h \cdot \rho \cdot g \cdot \dot{V}}{\eta_p} \quad (5.16)$$

Der Wirkungsgrad kann wieder abhängig von einem bekannten Wirkungsgrad η_0 bei der Drehzahl n_0 bestimmt werden [61, S. 22] [112]:

$$\eta_p = 1 - (1 - \eta_0) \cdot \left(\frac{n_0}{n}\right)^{0.1} \quad (5.17)$$

Hydraulische Modellierung von Wärmeübergabestationen

Der Massenstrom \dot{m} der Wärmeübergabestationen berechnet sich durch Umstellung der bereits bekannten Gleichung (3.1) für den Wärmestrom \dot{Q} [61, S. 23]:

$$\dot{m} = \frac{\dot{Q}}{c_p \cdot (T_{VL} - T_{RL})} \quad (5.18)$$

Dabei müssen die Wärmeabnahme \dot{Q} und die Rücklauftemperatur T_{RL} vorgegeben werden. Die Vorlauftemperatur T_{VL} ergibt sich an jeder Übergabestation aus der thermischen Simulation in Kapitel 5.2. Die Berechnung der Rücklauftemperatur T_{RL} wird ebenfalls dort erläutert. Für die Bestimmung der Wärmeabnahmen \dot{Q} werden Prognosen verwendet, welche zunächst die gesamte Netzlast bestimmen und anschließend die Last auf die verschiedenen Verbrauchertypen im Netz verteilen.

Die Berechnung der Druckverluste Δp über jede Wärmeübergabestation wird in den Modellen in dieser Arbeit nicht durchgeführt. Der Vorlaufdruck an jeder Station ergibt sich aus den durch Pumpen geregelten Vorlaufdrücken und Massenströmen sowie entsprechenden Druckverlusten in den Vorlaufleitungen. Der Rücklaufdruck wird anschließend an dem jeweiligen Netzschlechtpunkt so gewählt, dass ein bestimmter Differenzdruck eingehalten wird. Die Differenzdrücke an den anderen Wärmeübergabestationen ergeben sich aus der hydraulischen Berechnung der Rohrleitungen im Vor- und Rücklauf. Das hydraulische Verhalten der Wärmeübergabestationen wird also nicht berechnet. Es werden in jeder Simulation lediglich Mindestdifferenzdrücke an den Netzschlechtpunkten vorgegeben.

Soll das Verhalten der Stationen berücksichtigt werden, kann der Druckverlust mittels Gleichung (5.19) bestimmt werden [61, S. 24]:

$$\Delta p = \frac{1}{\rho \cdot d_{quiv}^4} \cdot \zeta_{WT} \cdot |\dot{m}| \cdot \dot{m} \quad (5.19)$$

d_{quiv} und ζ_{WT} stehen hier für den äquivalenten Durchmesser und den Widerstandsbeiwert eines Wärmetauschers und können durch Messwerte oder Modellierung bestimmt werden [61, S. 24].

Hydraulische Modellierung von Wärmeerzeugungsanlagen

Die Massenstromberechnung erfolgt wie bei Wärmeübergabestationen abhängig von der Leistung und der Differenz zwischen Vor- und Rücklauftemperatur nach Gleichung (5.18). Ein Druckverlust spezifisch durch eine Anlage kann auch wie bei Wärmetauschern durch Gleichung (5.19) berechnet werden.

Insgesamt muss von den Wärmeerzeugungsanlagen zusammen mit den Pumpstationen die Massenstrombilanz eingehalten werden. Der gesamte Massenstrom ergibt sich abhängig von den Abnahmen an den Wärmeübergabestationen, der jeweiligen (Differenz-) Druckregelung der Pumpen und den hydraulischen Verlusten durch das Rohrleitungssystem. Bei einer Anlage pro Netz ist der Massenstrom an dieser Anlage durch die Gegebenheiten im Netz also bereits festgelegt.

Bei thermisch stationären Simulationen ist die Temperaturlausbreitung im Netz nicht zeitlich variabel und die von Wärmeübergabestationen abgenommene Leistung muss zeitgleich ins Netz eingespeist werden. Bei einer Wärmeerzeugungsanlage ist dadurch deren Leistung eindeutig festgelegt. Bei thermisch instationären Rechnungen ist die Temperaturlausbreitung zeitlich abhängig. Die Temperaturen an den Wärmeübergabestationen, welche den Massenstrom im Netz maßgeblich beeinflussen, werden nicht sofort durch eine Änderung der Vorlauftemperatur an den Wärmeerzeugungsanlagen verändert. Somit kann die Vorlauftemperatur und Leistung einer Anlage höher oder niedriger als die Wärmeabnahme gewählt werden und das Netz dadurch temporär über- oder unterversorgt und als Speicher verwendet werden.

Gibt es mehrere Wärmeerzeugungsanlagen im Netz, können für alle Anlagen bis auf eine die Massenströme und mit den Vorlauftemperaturen auch die Leistungen vorgegeben werden. Die letzte Anlage muss dann die Massenstrombilanz erfüllen. Bei stationären Rechnungen ist damit auch die Leistung dieser Wärmeerzeugungsanlage definiert. Bei instationären Rechnungen können wiederum die Leistungen aller Anlagen durch Änderung der Vorlauftemperaturen variiert werden und müssen nicht mit der Wärmeabnahme im Netz übereinstimmen. Die Massenstrombilanz muss in diesem Fall eingehalten werden, die Leistungsbilanz allerdings nicht.

In einem Fernwärmenetz kann zum Beispiel abends die Temperatur abgesenkt werden. Die Temperaturen sinken dann langsam ab. Nachts sind die Temperaturen an den Wärmeübergabestationen entsprechend niedrig. Morgens kommt es allerdings zu Lastspitzen. Durch eine frühzeitige Erhöhung der Temperaturen an einer Wärmeerzeugungsanlage ändert sich zunächst an den Wärmeübergabestationen nichts. Der Massenstrom ins Netz bleibt gleich, die eingespeiste Leistung steigt durch die höheren Vorlauftemperaturen und wird im Netz gespeichert. Im besten Fall erreichen die höheren Vorlauftemperaturen die Wärmeübergabestationen, während morgens die Abnahme im Netz steigt. So kann der Massenstrom relativ konstant gehalten und die Leistungseinspeisung der Wärmeerzeugungsanlage muss nicht sprunghaft angehoben werden.

Kirchhoffsche Gesetze

Bei hydraulischen Netzberechnungen müssen die Kirchhoffschen Gesetze eingehalten werden. Diese stammen aus der Elektrotechnik, lassen sich aber durch Analogien auch auf hydraulische Netze anwenden. So werden zum Beispiel Wasser-, Gas- oder Fernwärmenetze mit ähnlichen Methoden berechnet [61, S. 28]. Anwenden lassen sich die Kirchhoffschen Gesetze bei durch Potenzialunterschiede (Spannung $\hat{=}$ Druck) bewegten Transportgrößen (Strom $\hat{=}$ Massenstrom), welche mittels eines Transportwiderstands (elektrischer $\hat{=}$ hydraulischer Widerstand) bestimmt werden.

1. Kirchhoffsches Gesetz, Knotenregel:

Die Summe der in einen Knotenpunkt fließenden Ströme entspricht der Summe der aus dem Knoten fließenden Ströme. An jedem Knotenpunkt im Netz gilt somit die Massenerhaltung. Entsprechend Abbildung 5.3 muss zum Beispiel gelten:

$$\dot{m} \stackrel{!}{=} \dot{m}_1 + \dot{m}_2 \quad (5.20)$$

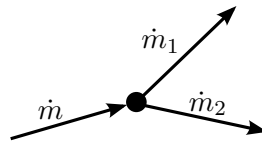


Abbildung 5.3.: Schematische Darstellung von Rohrleitungen für die Erläuterung des ersten Kirchhoffschen Gesetzes, der Knotenregel.

2. Kirchhoffsches Gesetz, Maschenregel:

Das zweite Kirchhoffsche Gesetz besagt, dass der Potenzialabfall entlang jeder Masche in Summe null ergeben muss. Bei elektrischen Netzen bezieht sich der Potenzialabfall auf die Spannung, bei Fernwärmenetzen auf den Druckverlust. Für die in Abbildung 5.4 dargestellte Masche muss also gelten:

$$\Delta p_1 + \Delta p_2 + \Delta p_3 + \Delta p_4 + \Delta p_5 \stackrel{!}{=} 0 \quad (5.21)$$

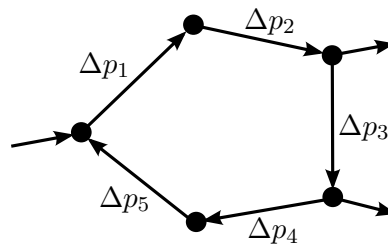


Abbildung 5.4.: Schematische Darstellung von Rohrleitungen für die Erläuterung des zweiten Kirchhoffschen Gesetzes, der Maschenregel.

5.2. Stationäre thermische Berechnung

Die stationäre thermische Rechnung wird ebenfalls nach Icking [61, S. 52 ff.] für die einzelnen Komponenten beschrieben. Die Bestimmung der Austrittstemperaturen der Komponenten T_A erfolgt jeweils in Abhängigkeit ihrer Eingangstemperaturen T_E nach Gleichung (5.22) [61, S. 65]:

$$T_A = C \cdot T_E + D \quad (5.22)$$

C und D sind dabei Koeffizienten, welche für die Komponenten im Folgenden definiert werden.

Thermische Modellierung von Rohrleitungen

Ein thermisches Modell für Rohrleitungen wird in [61, S. 53] vorgestellt. Das thermische Verhalten entspricht einem konvektiven, axialen Enthalpietransport. Der Wärmeübergang durch die Rohrwand ist radial. Für die Modellierung werden einige Annahmen getroffen:

- Radial ist die Strömung vollständig durchmischt, axial gibt es keine Rückvermischung. Die Wärmeleitung in axialer Richtung wird vernachlässigt.
- Der Wärmedurchgangskoeffizient zwischen Fluid und Umgebung ist unabhängig von Temperaturen und Strömungsgeschwindigkeiten.

Wird zusätzlich die Rohrreibung und die dadurch entstehende Reibungswärme vernachlässigt, ergibt sich die Ausgangstemperatur eines Rohres T_A folgendermaßen [61, S. 53]:

$$T_A = T_U + (T_E - T_U) \cdot e^{-\frac{4 \cdot U_R \cdot l}{d \cdot c_p \cdot \rho \cdot v}} \quad (5.23)$$

Dabei bezeichnet T_U die Umgebungstemperatur, T_E die Rohreingangstemperatur und U_R den Wärmedurchgangskoeffizienten. Bei Berücksichtigung der Reibungswärme kann Gleichung (5.23) erweitert werden [61, S. 53]:

$$T_A = T_U + (T_E - T_U) \cdot e^{-\frac{4 \cdot U_R \cdot l}{d \cdot c_p \cdot \rho \cdot v}} + \frac{v \cdot \Delta p \cdot d}{4 \cdot U_R \cdot l} \cdot \left(1 - e^{-\frac{4 \cdot U_R \cdot l}{d \cdot c_p \cdot \rho \cdot v}}\right) \quad (5.24)$$

$$\text{mit } C = e^{-\frac{4 \cdot U_R \cdot l}{d \cdot c_p \cdot \rho \cdot v}} \quad (5.25)$$

$$\text{und } D = T_U \cdot \left(1 - e^{-\frac{4 \cdot U_R \cdot l}{d \cdot c_p \cdot \rho \cdot v}}\right) + \frac{v \cdot \Delta p \cdot d}{4 \cdot U_R \cdot l} \cdot \left(1 - e^{-\frac{4 \cdot U_R \cdot l}{d \cdot c_p \cdot \rho \cdot v}}\right) \quad (5.26)$$

Maßgeblichen Einfluss auf die Temperaturänderung zwischen Rohreingang und -ausgang haben nach den Gleichungen (5.23) und (5.24) der Temperaturunterschied zwischen Fluid und Umgebung, der Wärmedurchgangskoeffizient U_R und die Strömungsgeschwindigkeit v . Die Bestimmung der Wärmedurchgangskoeffizienten ist laut [61, S. 54] schwierig und setzt genaue Kenntnis der verlegten Rohre, der Verlegeart und weiterer Daten voraus. Eine Anpassung der Koeffizienten durch Messungs- / Vergleichsrechnungen wird vorgeschlagen. Hier wird beispielhaft die Berechnung für einzeln verlegte Kunststoffmantelrohre mit parallel verlegtem Vor- und Rücklauf gezeigt. Nussbaumer [92, S. 127] erläutert die Berechnung der Wärmeübergangskoeffizienten nach Frederiksen [41] entsprechend Gleichung (5.27).

$$U_R = \frac{1}{\frac{r_R}{\lambda_d} \cdot \ln\left(\frac{r_M}{r_R}\right) + \frac{r_R}{\lambda_{Bo}} \cdot \ln\left(\frac{4 \cdot h}{r_M}\right) + \frac{r_R}{\lambda_{Bo}} \cdot \ln\left(\sqrt{\left(\frac{2 \cdot h}{s}\right)^2 + 1}\right)} \quad (5.27)$$

Die Werte λ_d und λ_{Bo} bezeichnen dabei die Wärmeleitfähigkeit des Dämmmaterials und des Erdreichs. r_R und r_M sind jeweils die Außenradien des Mediumrohres und des Rohres inklusive Wärmedämmung. Abbildung 5.5 zeigt eine schematische Darstellung der Rohre und verwendeten Einflussgrößen.

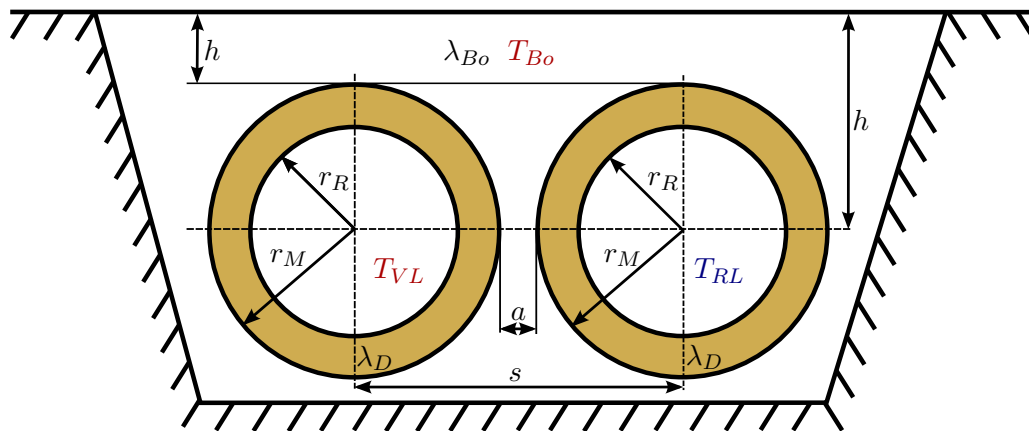


Abbildung 5.5.: Parallel verlegte Vor- und Rücklaufleitungen in Einzelrohrausführung nach [92, S. 127].

Die Verlegetiefe h wird mittels der minimalen Überdeckungshöhe r wie folgt berechnet:

$$h = h + r_M \quad (5.28)$$

Der Abstand der Mittelpunkte der beiden Rohre s berechnet sich nach Gleichung (5.29):

$$s = a + 2 \cdot r_M \quad (5.29)$$

In Gleichung (5.27) setzt sich die Berechnung des Wärmedurchgangskoeffizienten aus drei Teilen zusammen. Der Einfluss des Mediumrohres, der Wärmedämmung und der gegenseitige Einfluss der parallelverlegten Vor- und Rückläufe wird berücksichtigt. Nicht berücksichtigt wird zum Beispiel der Einfluss des Mantelrohres, welcher hier aus Kunststoff angenommen wurde und somit gering wäre. Auch Ablagerungen in den Rohren können die Wärmeübertragung verändern. Für die Berechnung von Stahlmantelrohren, welche wie in Kapitel 2.1.4 beschrieben eine zusätzliche Vakuumschicht besitzen, müsste die gegebene Formel (5.27) ebenfalls erweitert werden. In den hier entwickelten Modellen werden zunächst nur parallel verlegte Kunststoffmantelrohre berücksichtigt.

Thermische Modellierung von Ventilen

In einigen der hier entwickelten Modelle werden Ventile thermisch gleich den Rohrleitungen behandelt und somit lediglich ein Wärmeverlust berücksichtigt. Die Faktoren C und D entsprechen also denen von Rohrleitungen im vorherigen Kapitel. Ventile können allerdings auch folgendermaßen thermisch modelliert werden [61, S. 54]:

$$\Delta T = \frac{|\Delta p|}{\rho \cdot c_p} \quad (5.30)$$

$$\text{mit } C = 1 \quad (5.31)$$

$$\text{und } D = \frac{|\Delta p|}{\rho \cdot c_p} \quad (5.32)$$

Thermische Modellierung von Pumpen

Die Erwärmung des Fluids ΔT in einer Pumpe kann abhängig von der Druckdifferenz Δp und dem Wirkungsgrad η_p der Pumpe bestimmt werden [61, S. 54]:

$$\Delta T = |\Delta p| \cdot \frac{(1 - \eta_p)}{\eta_p \cdot \rho \cdot c_p} \quad (5.33)$$

$$\text{mit } C = 1 \quad (5.34)$$

$$\text{und } D = |\Delta p| \cdot \frac{(1 - \eta_p)}{\eta_p \cdot \rho \cdot c_p} \quad (5.35)$$

Thermische Modellierung von Wärmeübergabestationen

Der Wärmestrom an Wärmeübergabestationen ergibt sich nach Gleichung (3.1):

$$\dot{Q} = \dot{m} \cdot c_p \cdot (T_{VL} - T_{RL}) \quad (3.1)$$

Die Rücklaufftemperatur T_{RL} muss dabei vorgegeben werden. Dies kann durch Messdaten, Erfahrungswerte oder genaue Simulationen der Wärmeübergabestationen erfolgen. Auch die folgende Näherungsgleichung 5.36 kann verwendet werden [61, S. 56 f.]:

$$T_{RL} = T_{RL_0} + a \cdot f + b \cdot (T_{VL_0} - T_{VL}) \quad (5.36)$$

Der Auslastungsfaktor f beschreibt dabei das Verhältnis zwischen tatsächlicher Wärmeabnahme \dot{Q} und Nennanschlussleistung \dot{Q}_0 [61, S. 57]:

$$f = \frac{\dot{Q}}{\dot{Q}_0} \quad (5.37)$$

T_{RL_0} und T_{VL_0} bezeichnen die Auslegungsrücklauf- und Auslegungsvorlauftemperatur. Diese, sowie die Nennanschlussleistung \dot{Q}_0 und die Faktoren a und b sind Modellparameter jeder Wärmeübergabestation. Die vollständige Kalibrierung eines Netzmodells ist aufwändig und kann durch die Definition mehrerer Verbrauchergruppen mit gleichen Parametern vereinfacht werden [61, S. 57].

Bei den in dieser Arbeit verwendeten Modellen wird die Rücklaufftemperatur fest vorgegeben. Die Vorgabe der Rücklaufftemperaturen kann dabei auf Erfahrungswerten, Messwerten oder separaten Berechnungen basieren. Die Vorlauftemperatur ergibt sich aus der thermischen Berechnung.

Thermische Modellierung von Wärmeerzeugungsanlagen

Der Wärmestrom einer Wärmeerzeugungsanlage wird wie bei den Wärmeübergabestationen mit Gleichung 3.1 bestimmt. Die Rücklaufftemperatur T_{RL} ergibt sich dabei aus der thermischen Berechnung der Rohrleitungen im Rücklauf. Die Vorlauftemperatur muss vorgegeben werden.

Thermische Modellierung von Knotenpunkten

An Knotenpunkten im Netz treffen die Massenströme mehrerer Komponenten aufeinander und es gilt die Knotenbilanz, nach welcher die Summe aller in einen Knoten eintretenden und diesen Knoten verlassenden Wärmeströme null ergeben muss. Dabei wird der Knoten k_i mit den Komponenten v_{ij} betrachtet [61, S. 57]:

$$\sum \dot{Q}_{ij} = 0 \quad (5.38)$$

Bei der Modellierung wird davon ausgegangen, dass sich alle in einem Knoten treffenden Ströme sofort durchmischen [61, S. 57]. Die Ausgangstemperatur eines Knotens, in welchem sich mehrere Ströme treffen wird dann nach Gleichung (5.39) berechnet [61, S. 58]:

$$T_i = \frac{\sum_{\text{ein}} \dot{m}_{ij} \cdot T_{ij}}{\sum_{\text{aus}} \dot{m}_{ij}} \quad (5.39)$$

5.3. Instationäre thermische Berechnung

In der Arbeit von Icking [61, S. 88 ff.] werden mehrere instationäre Berechnungsverfahren verglichen und ein neuer Ansatz, das Gradientenverfahren entwickelt. Diese Modellierung wird auch in einigen der hier entwickelten Modelle verwendet und soll deshalb vorgestellt werden. Dabei wird lediglich auf das thermisch instationäre Verhalten der Rohrleitungen eingegangen. Die Berechnung der anderen im vorherigen Kapitel 5.2 beschriebenen Komponenten wurde nicht verändert. Abbildung 5.6 zeigt den Wärmetransport durch ein Volumenelement in einer Rohrleitung. \dot{Q}_k bezeichnet den konvektiven Wärmestrom entlang des Rohres, \dot{Q}_V den Wärmeverlust in radialer Richtung.

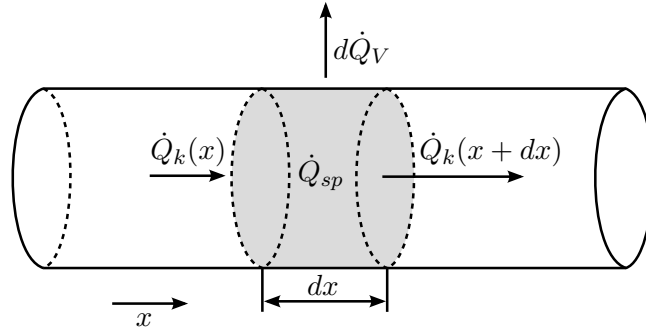


Abbildung 5.6.: Leistungsbilanz an einem Rohrelement nach [61, S. 89].

In der Arbeit von Icking [61, S. 88 ff.] wird aus der Leistungsbilanz (5.40) für ein infinitesimales Volumenelement die partielle Differenzialgleichung (5.41) für die Temperatur hergeleitet. A und U stehen für die Querschnittsfläche und den Umfang der Rohrleitung.

$$\frac{dQ_{sp}}{dt} = d\dot{Q}_k + d\dot{Q}_V \quad (5.40)$$

$$\frac{\partial T}{\partial t} = -\frac{\dot{m}}{\rho \cdot A} \cdot \frac{\partial T}{\partial x} - \frac{k \cdot U}{\rho \cdot A \cdot c_p} \cdot (T - T_U) \quad (5.41)$$

Für kleine Zeitschritte kann die Differenzialgleichung durch eine Differenzengleichung approximiert werden. Die Ausgangstemperatur eines Volumenelements aus einer Rohrleitung T_A kann nach Gleichung (5.42) abhängig von dessen Eingangstemperatur T_E sowie den Eintritts- und Austrittszeiten t_E und t_A berechnet werden [61, S. 98] [79, S. A-7]:

$$T_A = T_U + (T_E - T_U) \cdot e^{-\frac{4 \cdot U \cdot R_w}{\rho \cdot c_p \cdot d} \cdot (t_A - t_E)} \quad (5.42)$$

Die stationäre thermische Gleichung (5.23) wird also um die jeweilige Verweildauer des Fluids in einem Rohr erweitert. Als Austrittszeitpunkt t_A wird in einer Simulation der aktuelle Zeitschritt gewählt. Um die Austrittstemperatur T_A eines Volumenelements zu berechnen, wird in drei Schritten vorgegangen. Im ersten Schritt wird der Eintrittszeitpunkt t_E bestimmt. Die Bestimmung des Eintrittszeitpunktes und die dafür notwendigen Größen sind grafisch in Abbildung 5.7 dargestellt.

Das Integral der Strömungsgeschwindigkeit v von Eintritts- bis Austrittszeit muss dabei die Länge l der Rohrleitung ergeben [61, S. 94]:

$$\int_{t_E}^{t_A} v(t) dt = l \quad (5.43)$$

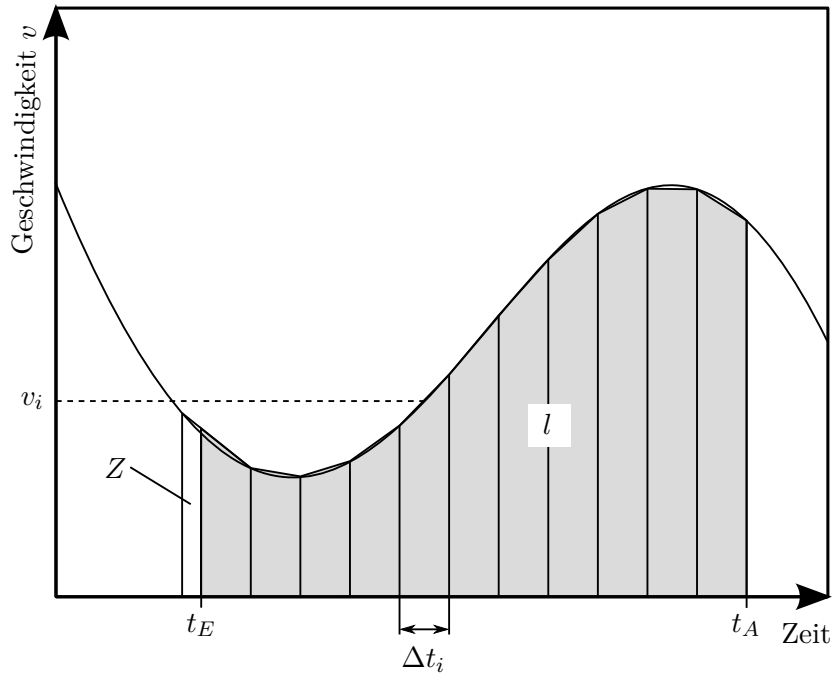


Abbildung 5.7.: Bestimmung des Eintrittszeitpunktes t_E nach [61, S. 95].

Bei einer Berechnung mit diskreten Zeitschritten muss numerisch integriert werden. Die Strömungsgeschwindigkeit v_i für einen Zeitschritt Δt_i wird dabei mit den Geschwindigkeiten der Zeitpunkte t_i und t_{i-1} gemittelt. Daraus ergibt sich für einen Zeitschritt die von dem Volumenelement zurückgelegte Strecke Δl_i :

$$\Delta l_i = v_i \cdot \Delta t_i \quad (5.44)$$

Die Strömungsgeschwindigkeiten sind für die thermische Berechnung notwendig und müssen im Voraus durch eine hydraulische Simulation bestimmt werden. Nun werden ausgehend vom Austrittszeitpunkt t_A so viele Zeitschritte zurückverfolgt, bis die gesamte Rohrlänge l erreicht oder überschritten wird. Der im letzten Zeitschritt zusätzlich zurückgelegte Weg Z wird wie folgt bestimmt [61, S. 95]:

$$Z = \sum_{i=1}^n (v_i \cdot \Delta t_i) - l \quad (5.45)$$

Der Eintrittszeitpunkt t_E eines Volumenelements wird durch Gleichung (5.46) ermittelt [61, S. 95].

$$t_E = t_A - \left(\sum_{i=1}^n \Delta t_i \right) + \frac{Z}{v_n} \quad (5.46)$$

Im zweiten Schritt kann die Eintrittstemperatur T_E bestimmt werden. Entspricht der Eintrittszeitpunkt t_E genau einem Zeitschritt, kann direkt die Temperatur dieses Zeitpunktes verwendet werden. Die Temperaturen vorheriger Zeitpunkte müssen für alle Knoten gespeichert werden. Ist der Eintrittszeitpunkt nicht genau an einem Zeitschritt, wird wieder zwischen zwei Zeitschritten interpoliert. Dies kann nach der Vorgehensweise in [61, S. 97 f.] durchgeführt werden. Im dritten Schritt wird die Austrittstemperatur T_A nach Gleichung (5.42) bestimmt.

Der Einfluss von Reibung auf die Temperaturentwicklung in den Rohrleitungen wird bisher nicht berücksichtigt. Auch der Einfluss der Wärmekapazität der Rohre auf die instationäre thermische Berechnung müsste berücksichtigt werden. Die entsprechenden Gleichungen werden in Icking [61, S. 99 ff.]

vorgestellt, werden hier aber nicht implementiert. Bei der vorgestellten Methode ist zu berücksichtigen, dass die Zeitschritte der Berechnung entsprechend auftretender Rohrlängen und Strömungsgeschwindigkeiten gewählt werden müssen. Diese Methode kann zum Beispiel kein Überspringen mehrerer Rohrleitungen durch ein Volumenelement berücksichtigen. Ist die Strömungsgeschwindigkeit so hoch, dass kurze Rohre in einem Zeitschritt bereits vollständig durchlaufen werden, muss der Berechnungsschritt kleiner gewählt werden. Bei großen, vollständig bis zu den letzten Wärmeübergabestationen abgebildeten Fernwärmenetzen führt dies zu sehr kleinen erforderlichen Zeitschritten und hohen Rechenzeiten.

Im Folgenden werden nun zwei Methoden der Zusammenführung der hydraulischen und thermischen Gleichungen in einem Simulationsmodell beschrieben.

5.4. Modellierung in *Simulink*

In *Simulink*[®] ist zu Beginn dieser Arbeit bereits ein Modell eines großen Fernwärmenetzes vorhanden. Das Modell verwendet eine stark manuell vereinfachte Netzstruktur und basiert auf den vorab vorgestellten hydraulischen und thermischen Gleichungen. Hydraulisch rechnet das Modell stationär und verwendet die Gleichungen für Rohrleitungen nach Kapitel 5.1. Nur der Rohrreibungsbeiwert wird nicht durch die *Colebrook-White*-Gleichung, sondern durch eine Schätzgleichung nach Böswirth [17] bestimmt. Thermisch werden Rohrleitungen entsprechend Kapitel 5.2 ohne Berücksichtigung der Reibungswärme modelliert. Ventile können für Regelungen eingesetzt werden und verursachen abhängig vom Öffnungswinkel einen Druckverlust. Der thermische Einfluss wird vernachlässigt. Die Pumpenkennlinie wird durch ein Polynom 2. Ordnung angenähert und Pumpen können direkt im Modell, zum Beispiel durch PID-Regler, die vorgegebenen Differenz- oder Rücklaufdrücke regeln. Thermisch werden auch Pumpen vernachlässigt. Zuletzt muss Wärmeverbrauchern ein Differenzdruck und eine Rücklauftemperatur sowie Wärmeeinspeisungen eine Vorlauftemperatur vorgegeben werden.

Beispielhaft wird hier das hydraulische Modell einer Rohrleitung in Abbildung 5.8 gezeigt. Zu sehen ist die grafische Zusammenführung der hydraulischen Gleichungen. Das Modell kann in einem Subsystem zusätzlich zu den thermischen Gleichungen für eine Rohrleitung als Komponente zusammengefasst werden. Die Parameter der Rohrleitungen wie beispielsweise Längen, Rohrleitungsbeiwerte oder Wärmeübergangskoeffizienten können von einem *MATLAB*[®]-Skript übergeben werden, müssen also nicht in jede Leitung einzeln eingetragen werden. Somit vereinfacht sich die Darstellung eines Fernwärmenetzmodells in *Simulink*[®]. Allerdings müssen nach wie vor alle Rohrleitungen einzeln in einer Oberfläche zusammengefügt werden sowie mit den relevanten Gleichungen für zum Beispiel die Mischtemperaturen an Knoten verbunden werden. Diese Art der Modellierung ist für große Fernwärmenetze sehr komplex. Ein Netz müsste zunächst stark aggregiert und vereinfacht werden. Dabei gehen Eigenschaften des Netzes verloren und entweder Drücke oder Temperaturen können nicht mehr genau modelliert werden. In Kapitel 3.1.1 wurden zum Beispiel die Aggregationsmethoden nach Löwen [79] und Larsen [75] vorgestellt, welche bei starker Aggregation respektive die Wärme- oder Druckverluste falsch berechnen.

Zudem ist zu beachten, dass das Modell nicht die Kirchhoffschen Gesetze erfüllt. Für die vollständige Beachtung der Gesetze müssen viele Rückführungen zwischen verschiedenen Wärmeverbrauchern, Leitungen und Knoten berücksichtigt werden, welche algebraische Schleifen verursachen. Diese können von *Simulink*[®] nur bedingt gelöst werden und führen zu hohen Rechenzeiten. Deshalb werden einige Vereinfachungen durchgeführt. Die Knotenregel (5.20) wird durch eine bestimmte Verschaltung der Massenströme zwischen verschiedenen Verbrauchern realisiert. In dieser Arbeit wurde zusätzlich getestet, bei mehreren aus einem Knoten gehenden Massenströmen konstante oder lastabhängige Faktoren zu verwenden, um die Massenströme auf die Leitungen zu verteilen und so Schleifen zu vermeiden. Die

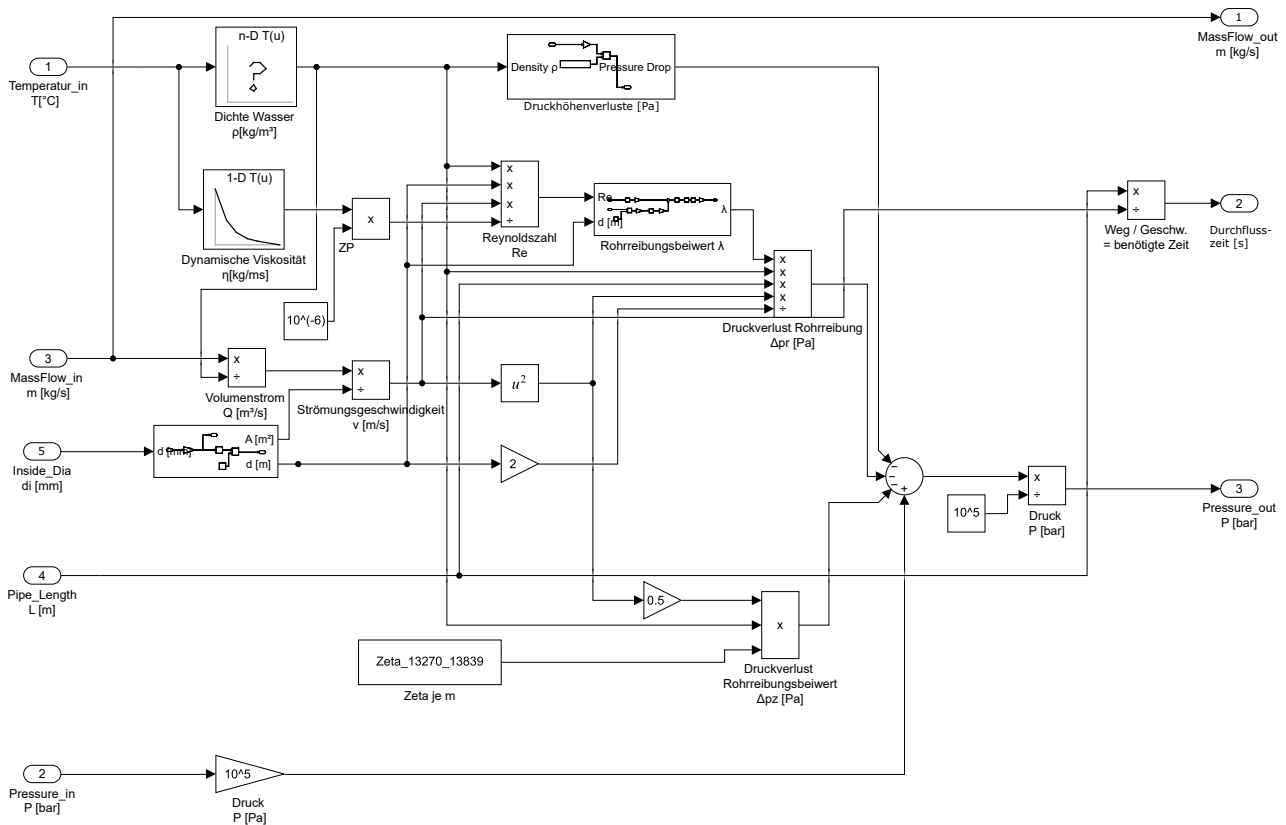


Abbildung 5.8.: Hydraulisches Modell einer Rohrleitung in *Simulink*[®].

Maschenregel (5.21) wird allerdings nicht beachtet. Das zu Beginn dieser Arbeit verfügbare Netzmodell wurde als Strahlennetz modelliert, weshalb in diesem Fall die Vernachlässigung der Maschenregel keine Nachteile verursacht. In der Modellierung wurde hauptsächlich darauf geachtet, die Drücke der Netzschlechtepunkte genau zu modellieren, um eine Differenzdruckregelung für das Netz auszulegen. Andere Knotenpunkte, vor allem im Rücklauf, weisen größere Abweichungen auf. Die in der vorliegenden Arbeit untersuchten Fernwärmenetze sind stärker vermascht und werden dementsprechend nicht so stark vereinfacht, dass Maschen komplett vernachlässigt werden können. Daher treten an Knoten, in welche mehrere Massenströme eintreten, auch größere Fehler bei der Druckberechnung auf.

5.5. Knoten-Kanten-Modelle

Als zweite Methode der Modellierung werden Knoten-Kanten-Modelle verwendet und in diesem Kapitel beschrieben. Zunächst werden in Kapitel 5.5.1 die als Basis verwendeten Knoten-Kanten- und Kanten-Maschen-Inzidenzmatrizen vorgestellt. Anschließend werden die hydraulische Netzberechnung in Kapitel 5.5.2, die thermische Netzberechnung in Kapitel 5.5.3 und deren Zusammenführung in Kapitel 5.5.4 ausgeführt. Dabei wird auch die hier entwickelte Modellierung in *MATLAB® R2020b* erläutert, welche auf den Arbeiten von Icking [61], Löwen [79] und Pierl [100] basiert. In dem Modell sind entsprechend den Anforderungen hydraulisch stationäre sowie thermisch stationäre und instationäre Berechnungen möglich. Zudem werden die kommerziellen Netzberechnungsprogramme *STANET®* und *SIR 3S®*, welche ebenfalls auf Knoten-Kanten-Modellen basieren, in Kapitel 5.5.5 präsentiert.

5.5.1. Knoten-Kanten-Matrix

Die vorherigen Kapitel 5.1 bis 5.3 haben die hydraulischen und thermischen Gleichungen der einzelnen Komponenten dargestellt. Durch die Graphentheorie kann aus den Komponenten eine sogenannte Knoten-Kanten-Matrix erstellt werden. Kanten repräsentieren in dieser Darstellung Verbindungselemente wie zum Beispiel Rohrleitungen, Ventile oder andere Komponenten, welche Knoten miteinander verbinden. Die Knoten markieren jeweils die Anfangs- und Endpunkte der Komponenten. An Knotenpunkten können auch mehrere Elemente zusammentreffen. Die Knoten-Kanten-Matrix beschreibt den Vor- und Rücklauf eines Fernwärmenetzes separat. Die Verbindung zwischen Vor- und Rücklauf erfolgt durch Wärmetauscher oder Wärmeerzeugungsanlagen. Pumpen werden ebenfalls nicht in der Knoten-Kanten-Matrix abgebildet.

Ein Graph $G = (\mathbf{K}, \mathbf{V})$ besteht aus einer Menge Knoten $\mathbf{K} = k_1, k_2, \dots$ und Kanten $\mathbf{V} = v_1, v_2, \dots$ [61, S. 25]. Fernwärmenetze werden als gerichteter Graph modelliert. Jede Kante verläuft also in eine definierte Richtung und beginnt in einem Knoten k_i sowie endet in einem Knoten k_j . Ein Kantenzug (v_1, \dots, v_n) ist Teil eines Graphen mit den Knotenpunkten k_0, \dots, k_n mit $v_i = (k_{i-1}, k_i)$ für $i = 1, \dots, n$ [61, S. 25]. Jede Kante kommt in einem Kantenzug nur einmal vor, die Knoten beliebig oft. Der Kantenzug wird dabei als Weg bezeichnet, wenn alle Knoten ebenfalls nur einmal vorkommen. Eine Masche ist ein Kantenzug mit demselben Start- und Endknoten. [61, S. 25 f.]

Ein Graph ist zusammenhängend, wenn es zwischen jedem Knotenpaar (k_i, k_j) einen verbindenden Weg gibt. Ein zusammenhängender Graph ohne Maschen wird als Baum bezeichnet. Sind alle Knoten in dem Baum enthalten und gibt es zwischen jedem Knotenpaar genau einen Weg, wird der Baum auch Gerüst genannt. Somit kann ein Strahlennetz zum Beispiel als Gerüst dargestellt werden. Aus einem vermaschten Netz lässt sich durch Entfernung einiger Kanten ebenfalls ein Gerüst erstellen. Bei einem Graphen mit k Knoten und v Kanten müssen m Kanten, auch Sehnen genannt, entfernt werden. Die Anzahl der Maschen des Graphen m ist folgendermaßen definiert [61, S. 26]:

$$m = v - k + 1 \quad (5.47)$$

Die Struktur des Netzes kann nun durch die Inzidenzmatrix \mathbf{A}_{inz} mit der Dimension $k \times v$ beschrieben werden. Für jeden Knoten k enthält die Matrix eine Zeile, für jede Kante v eine Spalte. Die Einträge der Matrix a_{ij} werden folgendermaßen beschrieben [61, S. 28]:

$$a_{ij} = \begin{cases} 1 & : \text{Flussrichtung der Kante } j \text{ zu Knoten } i \\ -1 & : \text{Flussrichtung der Kante } j \text{ aus Knoten } i \\ 0 & : \text{Kante } j \text{ und Knoten } i \text{ sind nicht verbunden} \end{cases} \quad (5.48)$$

Ein Netz kann ebenfalls als Kanten-Maschen-Inzidenzmatrix \mathbf{B}_{inz} mit der Dimension $m \times v$ beschrieben werden. m steht hier für die mittels Gleichung 5.47 ermittelte Anzahl der Maschen. Die Einträge der Matrix b_{ij} beschreibt [61, S. 28]:

$$b_{ij} = \begin{cases} 1 & : \text{ Kante } j \text{ und Masche } i \text{ verlaufen in die gleiche Richtung} \\ -1 & : \text{ Kante } j \text{ und Masche } i \text{ verlaufen in unterschiedliche Richtungen} \\ 0 & : \text{ Kante } j \text{ ist nicht in Masche } i \text{ enthalten} \end{cases} \quad (5.49)$$

Bei großen Fernwärmenetzen ist das Auffinden aller Maschen und das Bilden der Matrix \mathbf{B}_{inz} aufwändig. Algorithmen wie der in *MATLAB*[®] enthaltene *cyclebasis* können verwendet werden. Hier ist anzumerken, dass Maschen eigentlich nur einmal zu Beginn einer Berechnung gesucht werden müssen und die Kanten-Maschen-Inzidenzmatrix anschließend durchgängig so verwendet werden kann. Werden allerdings Ventile oder Schieber im Netz geschlossen und somit dessen Struktur verändert, muss die Matrix neu gebildet werden. Gleiches gilt für die Knoten-Kanten-Matrix, welche allerdings leichter aufzustellen ist. Ein einfaches Beispiel der hydraulischen Berechnung mittels einer Knoten-Kanten-Inzidenzmatrix und einer Kanten-Maschen-Inzidenzmatrix folgt in Kapitel 5.5.2 mit einem in Abbildung 5.9 dargestellten Netz.

5.5.2. Hydraulische Netzberechnung

Die stationäre hydraulische Netzberechnung erfolgt mit den Gleichungen aus Kapitel 5.1 und kann durch verschiedene Verfahren gelöst werden. Infrage kommen zum Beispiel die von Icking [61] beschriebenen Knoten- und Maschenverfahren. Im vorherigen Kapitel wurden die Knoten-Kanten- und Maschen-Kanten-Inzidenzmatrizen beschrieben und können verwendet werden, um die Anforderungen der Kirchhoffschen Gesetze aus Kapitel 5.1 zu formulieren.

Das erste Kirchhoffsche Gesetz besagt, dass alle Massenströme \dot{m} , die in einen Knoten fließen, auch wieder aus diesem herausfließen müssen. Dabei können Massenströme aus Kanten in die Knoten oder aus ihnen heraus fließen, welche interne Massenströme genannt werden. Zusätzlich können externe Massenströme \dot{m}_e Knoten hinzugefügt oder entnommen werden. Zu den externen Massenströmen zählen bei Fernwärmenetzen alle Massenströme zwischen dem Vor- und dem Rücklauf. Zum Beispiel wird an einer Wärmeerzeugungsanlage dem Rücklaufknoten ein externer Massenstrom entnommen und dem Vorlaufknoten hinzugefügt. An Wärmeübergabestationen wird entsprechend dem Vorlauf ein Massenstrom entnommen und dem Rücklauf hinzugefügt. Pumpen werden hier ebenfalls zu den externen Massenströmen gezählt. Das erste Kirchhoffsche Gesetz kann mit der Knoten-Kanten-Inzidenzmatrix \mathbf{A}_{inz} sowie den Vektoren der internen und externen Massenströme $\dot{\mathbf{m}}$ und $\dot{\mathbf{m}}_e$ folgendermaßen beschrieben werden [61, S. 29]:

$$\mathbf{A}_{inz} \cdot \dot{\mathbf{m}} = \dot{\mathbf{m}}_e \quad (5.50)$$

Für die Beschreibung des zweiten Kirchhoffschen Gesetzes wird die Maschen-Kanten-Matrix \mathbf{B}_{inz} und der Vektor der Differenzdrücke $\Delta \mathbf{p}$ verwendet. Das Gesetz besagt, dass alle Druckverluste entlang einer Masche in Summe null ergeben müssen [61, S. 30]:

$$\mathbf{B}_{inz} \cdot \Delta \mathbf{p} = 0 \quad (5.51)$$

Knotenverfahren

Das Knotenverfahren basiert auf der Gleichung für das erste Kirchhoffsche Gesetz (5.50). Für die Bestimmung aller internen Massenströme $\dot{\mathbf{m}}$ und Knotendrucke \mathbf{p} werden diese als Variablenvektor definiert:

$$\mathbf{x} = \begin{bmatrix} \mathbf{p} \\ \dot{\mathbf{m}} \end{bmatrix} \quad (5.52)$$

Dabei ist die Gleichung für den Differenzdruck (5.7) abhängig von der Widerstandsmatrix \mathbf{R} , dem Massenstrom und dem Höhenverlust. Druck und Massenstrom beeinflussen sich also gegenseitig.

$$\Delta \mathbf{p} = \mathbf{R}(\dot{\mathbf{m}}) \cdot \dot{\mathbf{m}} - \rho \cdot g \cdot \Delta \mathbf{h} \quad (5.7)$$

Die Widerstandsmatrix \mathbf{R} ist, wie in (5.6) beschrieben, ebenfalls von den Massenströmen $\dot{\mathbf{m}}$ abhängig. Die Elemente dieser Diagonalmatrix werden für jede Rohrleitung i folgendermaßen berechnet:

$$R_{i,i} = \frac{8}{\pi^2 \cdot d_i^4} \cdot \frac{1}{\rho_i} \cdot \left(\frac{l_i}{d_i} \cdot \lambda_i + \zeta_i \right) \cdot |\dot{m}_i| \quad (5.6)$$

Die Widerstandsmatrix wird von den Stoffwerten von Wasser in jeder Leitung und den Leitungen selbst beeinflusst. So ist sie von den Dichten ρ und den Rohrreibungsbeiwerten λ abhängig, welche wiederum von den Viskositäten μ abhängen. Für die Bestimmung dieser Stoffwerte müssen vor einer hydraulischen Berechnung die Temperaturen im Netz bekannt sein oder angenommen werden. Der Differenzdruck kann auch über die transponierte Knoten-Kanten-Inzidenzmatrix und die Knotendrucke bestimmt werden [61, S. 30]:

$$\Delta \mathbf{p} = -\mathbf{A}_{inz}^\top \cdot \mathbf{p} \quad (5.53)$$

Somit können Drücke und Massenströme mit folgendem Gleichungssystem (5.55) durch Kombination von (5.50), (5.7) und (5.53) gleichzeitig berechnet werden:

$$\mathbf{A}_{erw} \cdot \mathbf{x} = \mathbf{b}_{erw} \quad (5.54)$$

$$\begin{bmatrix} \mathbf{0} & \mathbf{A}_{inz} \\ \mathbf{A}_{inz}^\top & \mathbf{R}(\dot{\mathbf{m}}) \end{bmatrix} \cdot \begin{bmatrix} \mathbf{p} \\ \dot{\mathbf{m}} \end{bmatrix} = \begin{bmatrix} \dot{\mathbf{m}}_e \\ \rho \cdot g \cdot \Delta \mathbf{h} \end{bmatrix} \quad (5.55)$$

Dieses nicht-lineare Gleichungssystem kann nun iterativ als lineares Gleichungssystem nach \mathbf{x} bzw. nach \mathbf{p} und $\dot{\mathbf{m}}$ gelöst werden. Im ersten Schritt werden basierend auf Startwerten die erweiterten Matrizen \mathbf{A}_{erw} und \mathbf{b}_{erw} durch eine Berechnung der Stoffwerte und anschließend der Widerstandsmatrix \mathbf{R} , der externen Massenströme $\dot{\mathbf{m}}_e$ und der Höhenverluste bestimmt. Danach kann (5.55) als lineares Gleichungssystem gelöst werden. Da hier wieder neue Massenströme und Drücke berechnet werden, welche die Stoffwerte, Widerstandsmatrix, externen Massenströme und Höhenverluste beeinflussen, muss der Vorgang in einer Schleife ausgeführt werden, bis eine je nach Bedarf vorzugebende Genauigkeitsschranke eingehalten wird.

Nach Icking [61, S. 30 ff.] können Massenströme und Drücke auch separat berechnet werden. Somit kann die Anzahl der zu lösenden Gleichungen verglichen mit Gleichungssystem (5.55) reduziert werden. Der Massenstrom kann durch Umstellung von Gleichung (5.7) und Einsetzen von (5.53) für den Differenzdruck in Abhängigkeit des Drucks dargestellt werden:

$$\dot{\mathbf{m}} = \mathbf{R}(\dot{\mathbf{m}})^{-1} \cdot (-\mathbf{A}_{inz}^\top \cdot \mathbf{p} + \rho \cdot g \cdot \Delta \mathbf{h}) \quad (5.56)$$

Einsetzen dieser Gleichung (5.56) in Gleichung (5.50) ergibt ein neues, zu lösendes Gleichungssystem (5.57).

$$\mathbf{A}_{inz} \cdot \mathbf{R}(\dot{\mathbf{m}})^{-1} \cdot (-\mathbf{A}_{inz}^T \cdot \mathbf{p} + \boldsymbol{\rho} \cdot \mathbf{g} \cdot \Delta \mathbf{h}) = \dot{\mathbf{m}}_e \quad (5.57)$$

Das Gleichungssystem kann iterativ nach [61, S. 30 ff.] gelöst werden. Die Massenströme können dann mit Gleichung (5.56) berechnet werden.

In dieser Dissertation wird ein hydraulisches Knoten-Kanten-Modell basierend auf der Arbeit von Pierl [100] und den hier beschriebenen Gleichungen von Icking [61] in *MATLAB*[®] programmiert. In dem Modell wird das Gleichungssystem (5.55) durch eine Optimierung gelöst. Temperaturen werden dabei im ersten Schritt nach einem Referenzmodell in der Software *STANET*[®] oder einer thermisch stationären Berechnung vorgegeben. Anschließend kann thermisch stationär oder instationär weitergerechnet werden. Das Modell und dessen Einschränkungen werden im Folgenden erläutert.

Bei der Verwendung eines Optimierers für die Lösung des Gleichungssystems (5.54) $\mathbf{A}_{erw} \cdot \mathbf{x} = \mathbf{b}_{erw}$ können Nebenbedingungen verwendet werden. Bekannte Massenströme oder Drücke, zum Beispiel aus Messwerten oder an Festdruckknoten, können in einer Nachberechnung durch Gleichungsnebenbedingungen in folgender Form vorgegeben werden:

$$\mathbf{C}_{eq} \cdot \mathbf{x} - \mathbf{d}_{eq} = \mathbf{0} \quad (5.58)$$

Die Matrix \mathbf{C}_{eq} ist in diesem Fall der Teil der Matrix \mathbf{A}_{erw} , welcher nur die für die Nebenbedingungen relevanten Knoten und Kanten enthält und der Vektor \mathbf{d}_{eq} gibt die festen Drücke und Massenströme vor.

Durch Ungleichungsnebenbedingungen (5.59) können des Weiteren Minimal- oder Maximalwerte für bestimmte Massenströme oder Drücke im Vektor \mathbf{d}_{ineq} festgelegt werden. Hier können ebenso Differenzdrücke an Schlechtpunkten in Prognoserechnungen auf Mindestwerte festgelegt und so eine stationäre Differenzdruckregelung nachgebildet werden. Die Matrix \mathbf{C}_{ineq} enthält wieder den für die Nebenbedingungen relevanten Teil der Matrix \mathbf{A}_{erw} .

$$\mathbf{C}_{ineq} \cdot \mathbf{x} - \mathbf{d}_{ineq} \leq \mathbf{0} \quad (5.59)$$

In der Umsetzung in *MATLAB*[®] unter Verwendung eines Optimierers werden die Gleichungsnebenbedingungen (5.58) als Zielfunktion definiert. Das Gleichungssystem (5.55) wird dem Optimierer als Nebenbedingung und (5.59) als Ungleichungsnebenbedingung übergeben. Durch die Definition des hydraulischen Gleichungssystems (5.55) als Nebenbedingung wird dieses möglichst genau gelöst, während die Abweichungen zu bekannten Drücken und Massenströmen in der Definition von (5.58) als Zielfunktion nur minimiert werden.

Insgesamt werden in dem *MATLAB*[®]-Modell die Rohrleitungsgleichungen entsprechend Kapitel 5.1 berücksichtigt. Ventile werden als Kanten in der Knoten-Kanten-Inzidenzmatrix \mathbf{A}_{inz} behandelt und sind entweder vollständig geöffnet oder geschlossen. Andere Öffnungswinkel können bisher nicht berechnet werden. An Pumpen sind in Gleichung (5.55) externe Massenströme an ihren jeweiligen Knotenpunkten eingesetzt. Drehzahlen, Wirkungsgrade und Pumpenleistungen können anschließend wie in Kapitel 5.1 bestimmt werden. Pumpenkennlinien werden somit nicht direkt in der Lösung der Gleichungssysteme berücksichtigt. Auch Druckverluste von Wärmeübergabestationen und Wärmeerzeugungsanlagen werden vernachlässigt. Die beiden Komponenten werden als externe Massenströme

berücksichtigt. Die Knoten vor und hinter einer Wärmeerzeugungsanlage sind als Festdruckknoten definiert. Die minimalen Differenzdrücke an den Wärmeübergabestationen, welche die Schlechtpunkte im Netz beschreiben, werden mittels Ungleichungsnebenbedingungen (5.59) festgelegt.

Zur Verdeutlichung der vorgestellten Modellierung soll das Beispielnetz nach Abbildung 5.9 mit Knoten 1 bis 5 und Kanten *a* bis *e* im Vorlauf beschrieben werden. Die Knoten 6 bis 8 im Rücklauf an der Wärmeerzeugungsanlage und Pumpen sowie der restliche Rücklauf sind ebenfalls abgebildet.

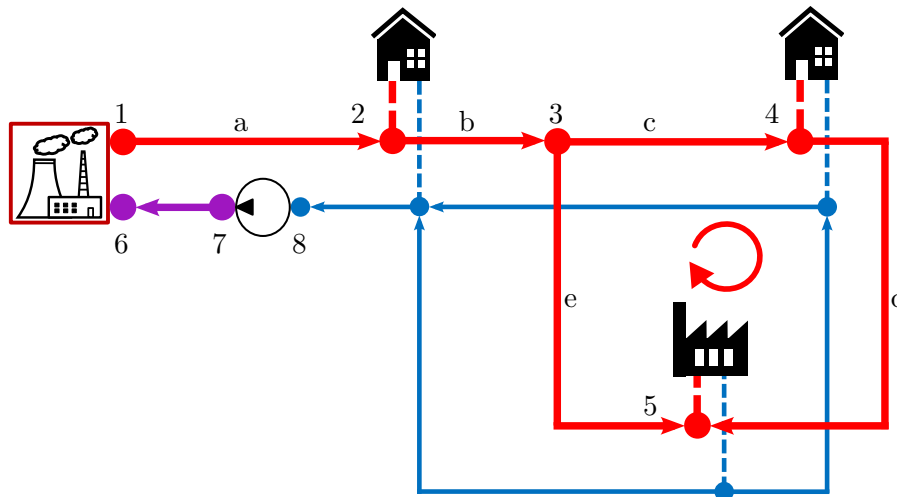


Abbildung 5.9.: Beispiel eines Fernwärmenetzes mit einer Masche und markierten Knoten und Kanten.

Das Netz kann als eine Knoten-Kanten-Inzidenzmatrix dargestellt werden, welche hier nur für den Vorlauf in Gleichung (5.60) gezeigt wird.

$$A_{inz,VL} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & -1 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \quad (5.60)$$

In dem Netz befindet sich zwischen Knoten 1 und 6 eine Wärmeerzeugungsanlage, welche dem Knoten 6 einen externen Massenstrom entnimmt und dem Knoten 1 mit einer festzulegenden Vorlauftemperatur hinzufügt. Die Knoten 1 und 6 sind zudem Festdruckknoten, ihr Druck muss vorab festgelegt werden. Bei den drei im Netz installierten Verbrauchern an den Knoten 2, 4 und 5 wird jeweils dem Vorlauf ein externer Massenstrom entnommen und dem Rücklauf wieder hinzugefügt. Temperaturen an allen Punkten im Netz müssen vor einer hydraulischen Berechnung gegeben sein. Zudem muss die gesamte Leistungsabnahme der Verbraucher vorab bestimmt und auf sie verteilt werden.

An der Umwälzpumpe wird Knoten 8 ein externer Massenstrom entnommen und Knoten 7 hinzugefügt. Die externen Massenströme der Pumpe und der Wärmeerzeugungsanlage müssen den insgesamt an den Verbrauchern entnommenen Massenströmen entsprechen, welche bei gegebener Temperaturverteilung und Leistungsabnahme bekannt sind. Bei mehreren Pumpen oder Verbrauchern muss deren gesamter Massenstrom ebenfalls dem Massenstrom der Verbraucher entsprechen. Dabei wird den Pumpen und Verbrauchern ein fixer Massenstrom vorgegeben. Eine Pumpe und ein Verbraucher müssen allerdings frei bleiben. Deren Massenströme ergeben sich aus der Berechnung der externen Massenströme in Gleichungssystem (5.55) und müssen die Massenstrombilanz erfüllen.

Für die Berechnung muss zusätzlich ein Druck im Rücklauf festgelegt werden. Hier wird der minimale Differenzdruck zwischen Vor- und Rücklauf der Optimierung als Nebenbedingung übergeben. Entspre-

chend werden am Schlechtpunkt des Netzes, in diesem Fall Knoten 5, der minimale Differenzdruck und somit ein Druck im Rücklauf festgelegt. Alle anderen Massenströme und Drücke resultieren aus der Lösung von Gleichungssystem (5.55). Hier ergibt sich eine weitere Einschränkung der hydraulischen Berechnung. Das Netz wird immer mit dem vorgegebenen minimalen Differenzdruck gefahren. In der Realität können Ausnahmen auftreten, wo dies nicht der Fall ist und tatsächlich höhere Differenzdrücke gefahren werden. Ist dies vorherzusehen, müssten in der Simulation ebenfalls höhere Differenzdrücke für die Schlechtpunkte festgelegt werden.

Maschenverfahren

Verbesserungen dieses hydraulischen Modells können zum Beispiel durch die separate Berechnung von Drücken und Massenströmen nach den bereits vorgestellten Gleichungen (5.57) und (5.56) erzielt werden. Eine weitere Möglichkeit der Reduktion der Rechenzeit ergibt sich durch das in Icking [61] vorgestellte Maschenverfahren. Beim Knotenverfahren müssen große Gleichungssysteme gelöst werden, welche von der Anzahl der Knoten k und Kanten v abhängen. Das Gleichungssystem (5.55) enthält $k + v$ Gleichungen, während bei der Berechnung nach (5.57) die Anzahl der zu lösenden Gleichungen bereits auf k reduziert wird. Das Maschenverfahren verringert die Anzahl weiter auf die Anzahl der Maschen m und wird ebenfalls in das *MATLAB*[®]-Modell integriert.

Die Maschen-Kanten-Inzidenzmatrix des Vorlaufs in Abbildung 5.9 lautet:

$$\mathbf{B}_{inz,VL} = [0 \quad 0 \quad 1 \quad 1 \quad -1] \quad (5.61)$$

Für die Anwendung des Maschenverfahrens muss das Fernwärmenetz im ersten Schritt in Gerüste und Sehnen entsprechend der Definition in Kapitel 5.5.1 getrennt werden. Das Maschenverfahren basiert auf Gleichung (5.51) für das zweite Kirchhoffsche Gesetz.

$$\mathbf{B}_{inz} \cdot \Delta \mathbf{p} = \mathbf{0} \quad (5.51)$$

Die Differenzdrücke $\Delta \mathbf{p}$ sind dabei abhängig von den Massenströmen $\dot{\mathbf{m}}$. Zunächst werden Startwerte für die Massenströme $\dot{\mathbf{m}}_0$ benötigt, welche entsprechend dem ersten Kirchhoffschen Gesetz bestimmt werden [61, S. 33]:

$$\mathbf{A}_{inz} \cdot \dot{\mathbf{m}}_0 = \dot{\mathbf{m}}_e \quad (5.62)$$

Für die Sehnen wird ein Startmassenstrom von 0 eingesetzt, für die Gerüste können die Massenströme dann direkt berechnet werden. Icking [61, S. 35 f.] leitet anschließend die weiteren für das Maschenverfahren notwendigen Gleichungen her. Gleichung (5.63) definiert die Berechnung der internen Massenströme $\dot{\mathbf{m}}$ basierend auf den Startwerten $\dot{\mathbf{m}}_0$, der transponierten Maschen-Kanten-Inzidenzmatrix \mathbf{B}_{inz}^\top und den Massenströmen in den Sehnen $\dot{\mathbf{m}}_S$ [61, S. 35].

$$\dot{\mathbf{m}} = \dot{\mathbf{m}}_0 + \mathbf{B}_{inz}^\top \cdot \dot{\mathbf{m}}_S \quad (5.63)$$

Gleichung (5.63) wird für die gesamte Berechnung in die vom Massenstrom abhängige Druckverlustgleichung (5.5) und das zweite Kirchhoffsche Gesetz (5.51) eingesetzt:

$$\Delta p(\dot{m}_i) = \frac{8}{\pi^2 \cdot d_i^4} \cdot \frac{1}{\rho_i} \cdot \left(\frac{l_i}{d_i} \cdot \lambda_i + \zeta_i \right) \cdot |\dot{m}_i| \cdot \dot{m}_i - \rho_i \cdot g \cdot \Delta h_i \quad (5.5)$$

$$\mathbf{B}_{inz} \cdot \Delta \mathbf{p}(\dot{\mathbf{m}}) = \mathbf{0} \quad (5.51)$$

$$\mathbf{B}_{inz} \cdot \Delta \mathbf{p}(\dot{\mathbf{m}}_0 + \mathbf{B}_{inz}^\top \cdot \dot{\mathbf{m}}_S) = \mathbf{0} \quad (5.64)$$

Dieses Gleichungssystem mit m Unbekannten entsprechend den Sehnenmassenströmen \dot{m}_S kann iterativ nach [61, S. 35] gelöst werden.

In dem *MATLAB*[®]-Modell sind ebenfalls zwei Methoden zur Simulation der Druck- und Differenzdruckregelung implementiert, die nachträgliche und die simultane Reglerkorrektur. Die Verfahren sind in der Arbeit von Icking [61, S. 41 ff.] beschrieben. Das Modell wird allerdings nur für eine Wärmeinspeisung programmiert. Die Umsetzung für mehrere Wärmeeinspeisungen kann zum Beispiel nach [61, S. 36 f.] erfolgen.

5.5.3. Thermische Netzberechnung

Die Implementierung stationärer und instationärer thermischer Berechnungen in Knoten-Kanten-Modellen wird im Folgenden erläutert.

Thermisch stationäre Netzberechnung

Für die hier vorgestellten thermischen Berechnungen werden die Massenströme als Eingangsgrößen vorausgesetzt. Zudem müssen Rücklauftemperaturen an den Wärmeübergabestationen und Vorlauftemperaturen an den Wärmeerzeugungsanlagen vorgegeben werden. Mit den bekannten Massenströmen können sogenannte „strömungsbezogene Kanten-Knoten-Inzidenzmatrizen“ [61, S. 62] gebildet werden. Die Elemente der Kanteneintritts-Knoten-Inzidenzmatrix S_E sind folgendermaßen definiert:

$$s_{E_{ij}} = \begin{cases} 1 : & \text{Strömung fließt von Knoten } j \text{ in Kante } i \\ 0 : & \text{Alle anderen Fälle} \end{cases} \quad (5.65)$$

(5.66) zeigt die Definition der Kantenaustritts-Knoten-Inzidenzmatrix S_A :

$$s_{A_{ij}} = \begin{cases} 1 : & \text{Strömung fließt von Kante } i \text{ in Knoten } j \\ 0 : & \text{Alle anderen Fälle} \end{cases} \quad (5.66)$$

Anschließend wird die folgende Gleichung (5.67) für den Vektor der n Knotentemperaturen T_k nach Icking [61, S. 66] gelöst. Die Diagonalmatrix W definiert dabei die Wasserwertströme, welche als Produkt aus Massenströmen und Wärmekapazitäten entsprechend Gleichung (5.68) berechnet werden [61, S. 64]. Die Koeffizienten C und D sind für die verschiedenen Komponenten in Kapitel 5.2 gegeben und werden hier als Diagonalmatrix bzw. Vektor verwendet.

$$T_k = (S_E \cdot W \cdot S_E^T - S_E \cdot W \cdot C \cdot S_E^T)^{-1} \cdot S_A \cdot W \cdot D \quad (5.67)$$

$$\text{mit } W = \text{diag}(\dot{m}_i \cdot c_{p,i}) \quad (5.68)$$

$$C = \text{diag}(C_i) \quad (5.69)$$

$$D = [D_1, D_2, \dots, D_n]^T \quad (5.70)$$

In der programmierten thermischen Modellierung werden Rohrleitungen und Ventile nach den Gleichungen für Rohrleitungen in Kapitel 5.1 berechnet. Für Pumpen wird keine Temperaturänderung berücksichtigt. Für Wärmeerzeugungsanlagen werden die Vorlauf-, für Wärmeübergabestationen die Rücklauftemperaturen vorgegeben. Die restlichen Temperaturen ergeben sich aus der thermischen Berechnung von Vor- und Rücklauf.

Thermisch instationäre Netzberechnung

Für die instationäre thermische Berechnung wird eine Knoteneintritts-Kanten-Inzidenzmatrix $\mathbf{S}_{E,inst}$ folgendermaßen gebildet:

$$s_{E_{ij},inst} = \begin{cases} 1 : & \text{Strömung fließt von Kante } j \text{ in Knoten } i \\ 0 : & \text{Alle anderen Fälle} \end{cases} \quad (5.71)$$

Diese Matrix basiert auf einer vorherigen hydraulischen Berechnung. Alle Flussrichtungen sind somit bekannt. Zusätzlich wird für jede Kante, hier für alle Rohrleitungen und Ventile, der jeweilige Anfangsknoten gespeichert. Anschließend wird für alle Elemente ein Temperaturkoeffizient berechnet. Dieser besteht jeweils aus der Berechnung der Ausgangstemperatur multipliziert mit dem Massenstrom. Für Leitungen und Ventile wird der Temperaturkoeffizient basierend auf Gleichung (5.42) folgendermaßen berechnet:

$$T_{coef,i} = \left(T_U + (T_{E_i} - T_U) \cdot e^{-\frac{A \cdot U_{R_i}}{\rho_i \cdot c_{p_i} \cdot d_i} \cdot (t_A - t_{E_i})} \right) \cdot \dot{m}_i \quad (5.72)$$

Als Austrittszeitpunkt t_A wird dabei der jeweilige Zeitschritt der Simulation verwendet. Der Eintrittszeitpunkt t_E wird für jede Leitung und jedes Ventil nach Gleichung (5.46) berechnet. Die Rohreintrittstemperatur T_E muss in jedem Zeitschritt wieder gespeichert werden und steht somit für zurückliegende Zeitpunkte zur Verfügung. Für Pumpen, Wärmeerzeugungsanlagen und Wärmeübergabestationen werden folgende Gleichungen für die Temperaturkoeffizienten verwendet:

$$\text{Pumpen : } T_{coef,i} = T_{E_i} \cdot \dot{m}_i \quad (5.73)$$

$$\text{Wärmeerzeugungsanlagen : } T_{coef,i} = T_{VL_i} \cdot \dot{m}_i \quad (5.74)$$

$$\text{Wärmeübergabestationen : } T_{coef,i} = T_{RL_i} \cdot \dot{m}_i \quad (5.75)$$

Für Pumpen werden also keine Temperaturänderungen berücksichtigt, die Eingangstemperatur T_E wird direkt weitergegeben. Für Wärmeerzeugungsanlagen muss die Vorlauftemperatur T_{VL} vorgegeben werden, für Wärmeübergabestationen die Rücklauftemperatur T_{RL} .

An jedem Knoten muss nun die Wärmestrombilanz nach Gleichung (5.39) eingehalten werden.

$$T_i = \frac{\sum_{\text{ein}} \dot{m}_{ij} \cdot T_{ij}}{\sum_{\text{aus}} \dot{m}_{ij}} \quad (5.39)$$

Die Gleichung für den Vektor der Knotentemperaturen \mathbf{T}_k lässt sich dann wie folgt mit dem Vektor aller Temperaturkoeffizienten \mathbf{T}_{coef} formulieren:

$$\mathbf{T}_k = \frac{\mathbf{S}_{E,inst} \cdot \mathbf{T}_{coef}}{\mathbf{S}_{E,inst} \cdot \dot{\mathbf{m}}} \quad (5.76)$$

5.5.4. Zusammenführung der hydraulischen und thermischen Netzberechnung

Bevor auf das Zusammenspiel der hydraulischen und thermischen Netzberechnung eingegangen wird, sollen die Eingangsgrößen für eine Simulation beschrieben werden. Die Werte wurden in den vorigen Kapiteln bereits erwähnt und sollen hier zur Verdeutlichung zusammengefasst werden. Diese Eingangswerte sind nicht nur für Knoten-Kanten-Modelle, sondern auch für andere Netzsimulationen erforderlich. Für die Wärmeerzeugung muss allen Erzeugungsanlagen im Modell eine Vorlauftemperatur vorgegeben werden. Ebenfalls muss deren Leistung oder Massenstrom gegeben sein. Dabei können allen Wärmeerzeugungsanlagen bis auf einer Anlage fixe Leistungen oder Massenströme vorgegeben werden. Die übrige Anlage speist den für die Massenstrombilanz notwendigen Massenstrom ein. Ihre Leistung ist somit variabel. Dieses Vorgehen entspricht der realen Fahrweise der Erzeugungsanlagen in Fernwärmenetzen. Die für Wärmeerzeugungsanlagen genannten Eingangswerte sind auch die Variablen, welche in dieser Arbeit vorrangig optimiert werden sollen.

Pumpen befinden sich oft in Wärmeerzeugungsanlagen und werden dort entsprechend der an die Anlagen vorgegebenen Massenströme eingestellt. Zusätzlich muss an einer Stelle im Netz eine Druckhaltung vorgegeben werden. Dies wird in dem *MATLAB*[®]-Modell durch Festdrücke an einer Wärmeerzeugungsanlage bewerkstelligt, kann aber auch durch eine modellierte Druckhaltung mit Pumpe erfolgen. Eine Umwälzpumpe im Netz stellt die Massenstrombilanz sicher und regelt Drücke im Netz. Dies kann zum Beispiel ein Rücklauf-, Vorlauf- oder Differenzdruck im System sein. Sollwerte sind entsprechend vorzugeben. Für Druckerhöhungsanlagen müssen Regelungslogiken implementiert oder fixe Massenströme bzw. Druckerhöhungen vorgegeben werden. Pumpenkennlinien und Informationen über Begrenzungen der Pumpen sind ebenfalls notwendig.

Rohrleitungen haben keine Eingangswerte für eine Simulation. Es sind aber alle hydraulischen und thermischen Parameter wie Rohrrauigkeiten oder Wärmedurchgangskoeffizienten für die Berechnung notwendig und können entweder entsprechend verfügbarer Informationen über die Leitungen oder durch Kalibrierung der Modelle gewählt werden. Für Ventile müssen die Widerstandsbeiwerte als Parameter und Öffnungswinkel als Eingangswerte angegeben werden. Öffnungswinkel können auch durch die Implementierung von Regelungen festgelegt werden. In dieser Arbeit werden die Regelungen von Pumpen und Ventilen nicht verändert und wie in den existierenden Fernwärmenetzen umgesetzt.

Des Weiteren sind die Wärmeabnahmen und Rücklauftemperaturen als Eingangswerte für die Wärmeübergabestationen notwendig. Diese Verbraucher bestimmen durch die gefragte Wärmeleistung maßgeblich die sich im Netz ausbreitenden hydraulischen und thermischen Zustände. Für eine Nachrechnung eines Zeitpunkts oder Tagesverlaufs können gemessene Wärmeabnahmen verwendet werden. Bei einer Prognoserechnung des Netzes ist auch eine Prognose der Wärmeabnahme notwendig. Hier kommen verschiedene Verfahren zum Einsatz, wie zum Beispiel Standardlastprofile, temperaturabhängige Profile oder der Aufbau von komplexeren Prognosemodellen, welche auf Regressionsverfahren oder Ansätzen des maschinellen Lernens wie beispielsweise neuronalen Netzen beruhen. Des Weiteren werden Wärmeabnahmen oft nicht kontinuierlich von allen Verbrauchern an die Netzbetreiber übermittelt. Bei bekannter gesamter Wärmeabnahme sind Annahmen der Wärmeverteilung im Netz zu treffen, welche entsprechend dem Anschlusswert der Abnehmer oder deren Klassifizierung zum Beispiel durch Standardlastprofile nach [12] erfolgen können. In der Arbeit von Pierl [100] wird zudem ein Ansatz der Abnahmeschätzung basierend auf Messdaten entwickelt. In dem hier implementierten *MATLAB*[®]-Modell basiert die Wärmeverteilung auf einem Hochlastfall aus *STANET*[®], in welchem alle Wärmeverbraucher einen hohen Verbrauch aufweisen. Für andere Lastfälle wird die Wärmeabnahme an allen Wärmetauschern durch Skalierung reduziert oder weiter erhöht und somit gleichmäßig im Netz verteilt. Eine andere Verteilung der Wärme durch Hinzukommen oder Wegfallen von Abnahmen an spezifischen Netzknoten wird nicht simuliert. Es besteht aber die Möglichkeit andere Methoden der Wärmeverteilung nachträglich zu implementieren.

Mit der Kenntnis der Wärmeabnahme und deren Verteilung im Netz können die Verbräuche in den simulierten Wärmetauschern eingestellt werden. Diese werden in vielen der recherchierten Modellimplementierungen nicht komplex modelliert und nur über Gleichung (5.18) berechnet. Aus der Temperaturdifferenz der Wärmetauscher und der Wärmeabnahme ergibt sich somit ein Massenstrom. Vorausgesetzt ist die Angabe der Rücklauftemperatur, welche für Nachrechnungen aus Messdaten entnommen oder für Prognosen approximiert werden kann. In Kapitel 3.3.1 wurde bereits erläutert, dass bei einer Vorlauftemperaturoptimierung der Einfluss auf die Rücklauftemperatur beachtet werden muss. Auch bei konstanten Vorlauftemperaturen beeinflusst die über den Tag schwankende Wärmeabnahme die Rücklauftemperaturen. Somit ist bei einer detaillierten Optimierung die Bestimmung dieser Temperaturen wichtig.

Bei der Durchführung einer Rechnung benötigt nun die hydraulische Simulation zunächst die Temperaturen und Wärmeverbräuche als Eingang. Andererseits benötigt die thermische Simulation Drücke und Massenströme. Daher muss hier iterativ gerechnet werden. In [93] und [61, S. 62] werden Temperaturen im ersten Schritt abgeschätzt und anschließend hydraulische und thermische Simulationen alternierend durchgeführt, bis eine vorzuziehende Genauigkeit eingehalten wird.

Für die stationäre thermische Simulation wird dieses Vorgehen so in dem *MATLAB*[®]-Modell umgesetzt. Für die instationäre Berechnung wird im ersten Zeitschritt für die Initialisierung eine stationäre thermische und hydraulische Iteration durchgeführt. In den folgenden Zeitschritten wird jeweils nach der hydraulischen Berechnung eine instationär thermische Rechnung durchgeführt und nicht weiter iteriert. Bei kleinen Zeitschritten ist die Temperaturänderung entsprechend gering und hat kaum Einfluss auf die hydraulische Berechnung. Somit ist eine häufige Iteration nicht unbedingt erforderlich.

5.5.5. Kommerzielle Programme

In dieser Arbeit wurden für die Modellierung auch die Programme *STANET*[®] vom Ingenieurbüro Fischer-Uhrig und *SIR 3S*[®] von der Firma 3S Consult GmbH verwendet. Beide Programme basieren auf Knoten-Kanten-Modellen und werden hydraulisch und thermisch ähnlich wie vorab beschrieben modelliert. Die Programme bieten für erstellte Fernwärmenetze die Option, stationäre und instationäre Simulationen durchzuführen. Zudem können Netze basierend auf GIS-Daten erstellt und durch bereits implementierte Algorithmen automatisiert aggregiert werden. Modelle können von anderen Programmen aus gestartet, die Variablen manipuliert und Ergebnisse ausgelesen werden. Dabei können Variablen jeweils in CSV-Dateien vor der Programmausführung verändert und Ergebnisse anschließend wieder in CSV-Dateien ausgelesen werden. *STANET*[®] kann über Batch-Dateien gestartet werden, während die Firma 3S Consult GmbH für *SIR 3S*[®] eine in *C#* geschriebene Schnittstelle für die Kompilierung zu einer EXE-Datei bereitstellt.

6. Modellbildung- und auswertung für vier Beispielnetze

Im vorherigen Kapitel wurde die thermo-hydraulische Modellierung von Fernwärmenetzen beschrieben, welche zum Beispiel mit Knoten-Kanten-Modellen umgesetzt werden kann. Die Anforderungen an ein Simulationsmodell, beschrieben in Kapitel 4.2.1, sind im Wesentlichen eine hohe Genauigkeit und Rechengeschwindigkeit, um eine Simulation und Optimierung der Fernwärmenetze in Echtzeit zu ermöglichen. Der Detaillierungsgrad der Abbildung eines Netzes soll hier noch ermittelt werden. Die vollständige Darstellung mit allen Rohrleitungen und Wärmeübergabestationen ermöglicht eine genaue Optimierung und Einhaltung aller Restriktionen, benötigt aber auch die höchste Rechenzeit. Somit ist unter Umständen eine Aggregation oder Vereinfachung der Netzmodelle nötig, um Restriktionen nur an den kritischen Punkten zu berechnen. Dazu zählen zum Beispiel die Druckschlechtepunkte, anhand welcher der Differenzdruck im Netz geregelt wird. Besonders an diesen Punkten ist die Einhaltung minimaler Drücke und Vorlauftemperaturen wichtig. Wie in Kapitel 4.2.2 beschrieben, soll die hydraulische Berechnung nur stationär erfolgen, während für die thermische Simulation stationäre oder instationäre Modellierungen infrage kommen. Ein großer Teil der in Kapitel 7 vorgestellten Optimierungen benötigt nur stationäre Modelle, während für eine detaillierte Optimierung aller Größen wie zum Beispiel der in Kapitel 3.3 beschriebenen Nutzung des Netzes als thermischer Speicher eine instationäre Berechnung erforderlich ist.

In Kapitel 6.1 wird als Ausgangspunkt der Untersuchungen Fernwärmenetz 1 verwendet und in Kapitel 6.1.1 für den Vergleich einiger der bisher vorgestellten Modellierungsansätze gewählt. Für dieses Netz sind bereits ein hydraulisch kalibriertes, stationäres Modell in der Software *STANET*[®] sowie eine große Menge an Messdaten vorhanden. Das Netz wird auch in der Arbeit von Pierl [100] hydraulisch stationär in *C++* modelliert und für die Leckageortung verwendet. Zudem ist das Netz mit nur einer zentralen Einspeisung ein einfaches Beispiel. Ein vollständiges und ein aggregiertes thermo-hydraulisches Netzmodell werden in Kapitel 6.1.2 beschrieben. Es wird klar, dass aufgrund hoher Rechenzeiten weitere Vereinfachungen notwendig sind, auf welche im Folgenden detailliert eingegangen wird. Die thermo-hydraulischen Modelle dienen dabei als Basis, um Simulationsdaten zu erzeugen. Für rein stationäre Simulationen werden in Kapitel 6.1.3 Ansätze des maschinellen Lernens verglichen. Die Erstellung und Parameterwahl eines möglichst genauen Modells mit geringen Rechenzeiten werden untersucht und eine Ergebnisauswertung durchgeführt. Dabei wird auch auf den Einfluss der Aggregation des Fernwärmenetzes eingegangen. Da für einige Optimierungen thermisch instationäre Simulationen notwendig sind, wird in Kapitel 6.1.4 die Approximation instationärer Modelle getestet.

Fernwärmenetz 2 stellt mit zwei Wärmeerzeugungsanlagen und einem größeren Rohrleitungsnetz die nächste Komplexitätsstufe dar. Kapitel 6.2.1 beschreibt ein thermo-hydraulisches Simulationsmodell, welches ebenfalls aggregiert wird. Vorrangig wird wieder die Vereinfachung stationärer Modelle durch den Vergleich von Algorithmen des maschinellen Lernens in Kapitel 6.2.2 untersucht. So sollen die Ergebnisse von Fernwärmenetz 1 bestätigt werden. Anschließend wird ein Modellierungsansatz gewählt und im Detail ausgearbeitet. Zudem stellt sich heraus, dass die Verwendung der Rücklauftemperatur als Modelleingang Simulationsergebnisse deutlich verbessert. Eine Vorhersage der für eine Prognose-simulation nicht bekannten Rücklauftemperatur wird deshalb in Kapitel 6.2.3 vorgestellt.

Fernwärmenetz 3 ist das größte in dieser Arbeit behandelte Netz mit vier Wärmeerzeugungsanlagen inklusive Geothermieanlagen. In Kapitel 6.3.1 wird zunächst wieder ein thermo-hydraulisches Modell vorgestellt. Die für Fernwärmenetz 1 und 2 gesammelten Erkenntnisse bezüglich der Erstellung vereinfachter stationärer Modelle sollen anschließend in Kapitel 6.3.2 an dem komplexeren Fernwärmenetz 3 mit verteilter Wärmeeinspeisungen verifiziert werden. Fernwärmenetz 4 wird in Kapitel 6.4 grundlegend zusammengefasst. Da hier keine neuen Erkenntnisse gewonnen werden, wird die Erstellung von Simulationsmodellen in Anhang A.1.5 beschrieben. Abschließend werden die Ergebnisse der Modellbildung in Kapitel 6.5 diskutiert.

6.1. Fernwärmenetz 1

Die Architektur von Fernwärmenetz 1 ist schematisch in Abbildung 6.1 dargestellt. Zu sehen ist HKW 1 als Wärmeerzeugungsstandort. Zudem besteht eine Zuleitung von einem zweiten HKW, von welchem aus ebenfalls Wärme am gleichen Standort eingespeist werden kann. Für eine flexiblere Einspeisung sind am Standort des HKW 1 zwei Speicher vorhanden. Zwischen den zwei HKWs und den zwei Speichern kann die Wärmeeinspeisung optimiert werden. Die Vorlauftemperatur ergibt sich aus den Wärmeströmen dieser vier Anlagen und kann zusätzlich durch eine Beimischung aus dem Rücklauf abgekühlt werden. Neben der Vorlauftemperatur haben der Vorlaufdruck am Kraftwerksausgang sowie die Wärmeabnahme einen Einfluss auf das Netzverhalten. Eine Druckerhöhungsanlage (DEA) kann den Druck in weit von der Wärmeerzeugung entfernten Gebieten bei Bedarf erhöhen.

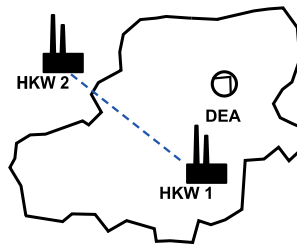


Abbildung 6.1.: Vereinfachte, nicht-maßstäbliche Darstellung des realen Fernwärmenetzes 1 mit Druckerhöhungs- und Wärmeerzeugungsanlagen.

6.1.1. Vorauswahl der Modellierungsansätze

Fernwärmenetz 1 wird nun als Grundlage für den Vergleich von thermo-hydraulischen Modellierungsansätzen verwendet. Das vorhandene Modell in *STANET*[®] wird dafür zunächst stark aggregiert und etwa zwei Drittel der Wärmeabnahmen werden gänzlich entfernt. So kann die Größe des Netzes von über 8000 Leitungen und Knoten auf 60 Leitungen und 58 Knoten reduziert werden und der Aufwand bei der Modellbildung sowie die Simulationszeiten gehen deutlich zurück. Das vereinfachte Netz mit Darstellung der übrigen Rohrleitungen ist in Abbildung 6.2 zu sehen. Mit diesem Netz werden anschließend drei Modellierungsansätze im Vergleich zu dem Referenzmodell in *STANET*[®] untersucht. Zuerst wird das Netz basierend auf dem *MATLAB*[®] *Simulink*[®]-Ansatz in Kapitel 5.4 modelliert. Für das kleine Modell fällt dabei bereits auf, dass die Implementierung der Netze in *Simulink*[®] ohne Automatisierung aufwändig ist. Alle Komponenten müssen manuell in der graphischen Oberfläche erstellt und verbunden werden. Die Darstellung eines vollständigen Netzes, wie zum Beispiel von Netz 1 mit etwa 8000 Knoten und Rohrleitungen ist nur unter erheblichem Aufwand möglich. Dazu kommt die Problematik der Modellierung von Maschen, siehe Kapitel 5.4, welche in *Simulink*[®] zu hohen Rechenzeiten führt. Ein Rechner mit *Intel Core i7-6700HQ* und 16 GB RAM benötigt eine Rechenzeit von etwa 350 s

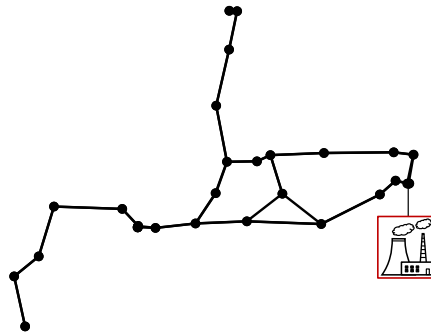


Abbildung 6.2.: Aggregiertes und reduziertes Fernwärmenetz 1.

für eine Tagessimulation des Fernwärmenetzes in *Simulink*[®]. Dabei müssen die Zeitschritte der Simulation variabel gewählt werden, da sonst Probleme mit algebraischen Schleifen auftreten. Weiterhin wird das thermische Verhalten in *Simulink*[®] nur instationär durch Zeitverzögerungen modelliert. Für einen Vergleich mit stationären *STANET*[®] Berechnungen werden mehrere Simulationen durchgeführt und das *Simulink*[®] Modell jeweils in einen eingeschwungenen stationären Zustand überführt. Für Drücke ergeben sich relative Abweichungen unter 3 %, während Massenströme im Durchschnitt um etwa 6 % abweichen. Allerdings treten auch höhere Abweichungen von bis zu 20 % in einigen Leitungen auf.

Da in den Recherchen in Kapitel 3.1 und 3.3 viele Projekte die Software *Modelica* verwenden, wurde parallel auch dieser Modellierungsansatz untersucht. Die Projekte kamen allerdings zu keinen brauchbaren Ergebnissen, sodass die Modellierung mit dieser Software verworfen wurde.

Hauptsächlich werden Modellierungsarbeiten mit den Knoten-Kanten-Modellen aus Kapitel 5.5 in *MATLAB*[®] *R2020b* durchgeführt. Für die Modellierung des aggregierten Fernwärmenetzes kann die Netzarchitektur aus dem vorhandenen *STANET*[®] Modell in *MATLAB*[®] importiert werden. Die Daten für die verschiedenen Komponenten, also Rohrleitungen, Ventile, Wärmeeinspeisung, Pumpen und Wärmeübergabestationen werden automatisiert in für die Berechnung notwendige Matrizen transformiert. So gestaltet sich die Modellerstellung deutlich schneller als in *Simulink*[®]. Andererseits ist für große Netze immer noch ein großer Aufwand für die Kalibrierung der Netzparameter notwendig. Zeitschritte können für eine Simulation festgelegt und sowohl stationäre als auch instationäre thermische Simulationen durchgeführt werden. Für beide Varianten wird ein Zeitschritt von 15 s gewählt.

Für die hydraulische Netzberechnung wird in *MATLAB*[®] ein Vergleich zwischen den in Kapitel 5.5.2 vorgestellten Knoten- und Maschenverfahren durchgeführt. Zudem werden die in *MATLAB*[®] vorhandenen Optimierer *lsqin* und *fmincon* mit der Optimierungssoftware *WORHP* verglichen. Mit dem Knotenverfahren gelöst durch *WORHP* und einer instationären thermischen Berechnung können durchschnittliche 24-Stunden-Druckabweichungen unter 1,5 % und Massenstromabweichungen unter 4,5 % für einen simulierten Beispieltag erreicht werden. Das Maschenverfahren erreicht Druckabweichungen unter 0,5 % und Massenstromabweichungen unter 1,5 %. Die Simulationszeiten liegen für das Knotenverfahren bei 3,2 s und für das Maschenverfahren bei 4 s. Thermisch stationär kann eine Simulation in etwa 1,7 s mit dem Maschenverfahren durchgeführt werden. Die durchschnittlichen Abweichungen liegen für Drücke bei 0,3 % und für Massenströme bei 0,8 %.

Insgesamt können mit den Ansätzen der Knoten-Kanten-Modelle genauere Ergebnisse bei geringeren Rechenzeiten und einer einfacheren Modellierung erreicht werden. Der *Simulink*[®] Ansatz soll somit nicht weiter zum Einsatz kommen. Auch dieser Ansatz kann zum Beispiel durch eine verbesserte Modellierung von Maschen und durch eine automatisierte Erstellung der Netzarchitektur verbessert werden, wird hier aber nicht weiter verfolgt.

Das Knoten-Kanten-Modell in *MATLAB*[®] wird anschließend für das vollständige Netz 1 erstellt und getestet. Die Abweichungen zu *STANET*[®] sind dabei etwas höher als für das aggregierte Netzmodell.

Die Rechenzeiten liegen bei etwa 30 s für thermisch stationäre Berechnungen und 800 s für thermisch instationäre Simulationen. Somit sind die Modelle für den Einsatz in einer MPC ungeeignet. In der Modellierung in den vorherigen Kapiteln wurden ebenfalls einige Berechnungen vernachlässigt, welche hier noch einmal zusammengefasst werden. In der Umsetzung in *MATLAB*[®] werden folgende hydraulisch relevante Berechnungen nicht berücksichtigt:

- Ventile werden nur geöffnet oder geschlossen betrachtet und als Rohrleitung modelliert. Verschiedene Ventilöffnungswinkel und die Implementierung von Ventilen in Regelungen sind bisher nicht möglich.
- Der Druckunterschied durch Wärmetauscher in Wärmeerzeugungsanlagen wird nicht berücksichtigt. Vor- und Rücklaufknoten der Anlagen sind Festdruckknoten.

Zudem werden einige thermische Einflüsse nicht modelliert:

- Der Wärmedurchgangskoeffizient wird nur für Kunststoffmantelrohre beschrieben. Weitere Rohrtypen müssen hinzugefügt werden.
- Ventile werden wie Rohrleitungen modelliert und nicht entsprechend Kapitel 5.2.
- Pumpen werden thermisch nicht modelliert.
- Bei der instationären thermischen Berechnung fehlen Reibungswärme und der Einfluss der Wärmekapazität der Rohre. Beides wird in [61] beschrieben.

Netze können des Weiteren nur aus *STANET*[®] Daten und nicht aus Geoinformationssystem (GIS)-Daten importiert werden. Auch Wärmeverbräuche werden nur basierend auf einem Lastfall aus *STANET*[®] auf die Wärmetauscher verteilt.

Insgesamt können mit dem *MATLAB*[®] Knoten-Kanten-Modell verschiedene Fernwärmenetze modelliert und simuliert werden. Das Beispielnetz 1 zeigt im Vergleich zu *STANET*[®] sehr gute Berechnungsergebnisse und die Modellierung einschließlich bisher nicht berücksichtigter Effekte wurden in dieser Arbeit vollständig beschrieben. Allerdings ist auch mit diesem Ansatz für die vollständige Modellierung und Kalibrierung der Fernwärmenetzmodelle ein großer Aufwand notwendig, weshalb entschieden wird, die weitere Modellierung mit kommerzieller Software durchzuführen, welche eine schnellere Modellierung der Netze unter Berücksichtigung aller hydraulischen und thermischen Effekte ermöglicht. Die Modellierung in diesen Programmen basiert ebenfalls auf der in Kapitel 5 vorgestellten Theorie. Infrage kommen in dieser Arbeit die beiden Programme *STANET*[®] und *SIR 3S*[®], welche bereits für die Modellierung der vier Fernwärmenetze im Einsatz sind. In *STANET*[®] ist das Netz 1 vollständig modelliert und hydraulisch validiert. Hier ist somit ein guter Stand vorhanden. Auch für die weiteren Netze sind Modelle verfügbar. Allerdings sind diese Modelle nicht thermisch kalibriert und validiert. Zudem müssen für die weiteren Netze der Modellierungsstand überprüft und die Modelle voraussichtlich angepasst werden. *STANET*[®] wird des Weiteren noch nicht für Online-Netzsimulationen und Regelungen in Fernwärmenetzen angewandt, was bei *SIR 3S*[®] der Fall ist. Die Software wurde zum Beispiel in Zusammenhang mit den Optimierungsprojekten um die Software *BoFiT* Anfang der 2000er-Jahre eingesetzt, siehe Kapitel 3.3.2. Aus diesen Gründen und der Verfügbarkeit der Firma 3S Consult GmbH für die Modellierung und Validierung der weiteren Netze fällt die Wahl auf die Software *SIR 3S*[®].

6.1.2. Netz 1: *SIR 3S*[®]

Die Modellierung der Netze in *SIR 3S*[®] wird von der Firma 3S Consult GmbH durchgeführt. Anschließend werden die Modelle im Austausch validiert sowie die Schnittstellen für die Steuerung und Auswertung der Modelle spezifiziert. Zunächst wird in *SIR 3S*[®] ebenfalls das vollständige Netz 1 mit

etwa 8000 Knoten und Rohrleitungen sowie den Stationen in Abbildung 6.1 modelliert. Die Länge der Rohrleitungen im Vorlauf beträgt etwa 100 km bei ca. 800 angeschlossenen Wärmeübergabestationen. Durch eine Aggregation auf ca. 1800 Knoten und 1400 Rohrleitungen reduziert sich die Leitungslänge auf etwa 38 km und die Anzahl der Wärmeübergabestationen auf 300. Das HKW ist sowohl erweitert mit Speichern, Zuleitung und Beimischung aus dem Rücklauf modelliert als auch vereinfacht nur mit einer Wärmeeinspeisung. Das Modell wird anhand von vier Beispieltagen mit Testdaten validiert:

- Tag 1: Hochlastfall im Januar 2019
- Tag 2: Fall in der Übergangszeit im April 2019
- Tag 3: Niedriglastfall im Juli 2019
- Tag 4: Fall in der Übergangszeit im Mai 2019

Für einen Simulationstag können mehrere stationäre Simulationen aneinandergereiht (quasi-stationär) oder eine instationäre Simulation durchgeführt werden. Bei der instationären Simulation wird nur die Temperaturausbreitung im Netz instationär berechnet. Druck und Massenstrom werden in beiden Möglichkeiten für jeden Zeitpunkt stationär simuliert. Druckstöße werden somit nicht berechnet. Dies ist hier nicht notwendig, da keine Druckregelungen ausgelegt werden sollen. Insgesamt ergeben sich als Eingangsvariablen des Netzmodells mit vereinfachter Stationsmodellierung die folgenden Größen:

- Wärmeabnahme
- Vor- und Rücklauftemperatur am HKW
- Vorlaufdruck
- Druckerhöhung der DEA

Die Rücklauftemperatur ist für Prognoseberechnungen nicht bekannt und kann theoretisch vorhergesagt werden, um die Netzberechnung zu verbessern. Ebenso kann sie für die Berechnung vernachlässigt und auf einen konstanten Durchschnittswert gesetzt werden. Die Validierung der *SIR 3S*[®] Modelle mit Messdaten von den vier Beispieltagen wird mit konstant gesetztem Vorlaufdruck und durchschnittlichen konstanten Rücklauftemperaturen durchgeführt. Diese Eingangsgrößen müssen folglich auch später in einer Optimierung nicht angepasst werden.

Die Simulationszeiten für stationäre 24-Stunden-Simulationen mit 15-Minuten-Zeitschritten des aggregierten Fernwärmenetzes liegen im Bereich weniger Sekunden bis zu einer Minute, instationäre Simulationen benötigen höhere Rechenzeiten. Tagen mit hohem Wärmeverbrauch können schneller berechnet werden als Tage mit niedrigem Wärmebedarf, an welchen es länger dauert, bis die Lösung der hydraulischen und thermischen Gleichungssysteme konvergiert. Dies liegt unter anderem an geringen Massenströmen und während der Iterationen wechselnden Flussrichtungen in den Maschen.

Die Modelle der Fernwärmenetze sollen im Anschluss für eine Optimierung und in einer MPC zum Einsatz kommen. Für diesen Zweck sind geringe Rechenzeiten im Bereich von Millisekunden notwendig, welche selbst durch die mit aktuellen Methoden programmierten physikalischen Knoten-Kanten-Modelle wie *SIR 3S*[®] auch bei Reduzierung des Prädiktionshorizonts auf unter 24 Stunden, beispielsweise auf vier oder sechs Stunden, nicht erreicht werden. Zwar kann die Rechenzeit durch Aggregation der Netze reduziert werden, allerdings ist dies nur bis zu einem gewissen Punkt möglich. Auch die Genauigkeit der Modelle wird von der Aggregation beeinflusst und nimmt mit zunehmendem Aggregationsgrad ab. Bisher wurde zudem noch nicht auf die Problematik der Berechnung von Gradienten eingegangen. In den Kapiteln 2.2.4 und 2.2.5 wurden Algorithmen für die lokale Optimierung beschrieben, welche in der Regel die ersten und/oder zweiten Ableitungen der Zielfunktion sowie Restriktionen benötigen. Mit den vorgestellten physikalischen Modellen ist eine Formulierung der Ableitungen je nach Zielfunktion schwierig. Somit sind insgesamt eine Vereinfachung und möglichst eine Formulierung der

Modelle, welche eine einfache Berechnung von Gradienten der verschiedenen Zustände ermöglicht, für schnelle Optimierungszeiten notwendig. Das thermo-hydraulische Modell in *SIR 3S*[®] bietet dafür eine validierte Grundlage.

6.1.3. Netz 1: Vereinfachte stationäre Modelle

Als Möglichkeit der Modellvereinfachung werden im Folgenden die in den Kapiteln 2.4 und 2.5 vorgestellten Ansätze der Systemidentifikation und des maschinellen Lernens untersucht. In beiden Fällen wird versucht, einen mathematischen Zusammenhang zwischen relevanten Ein- und Ausgangsdaten der Fernwärmenetze herzustellen, welcher eine schnelle Simulation der Netze ermöglicht. Eingangsdaten sind beispielsweise die Einspeiseleistungen und Vorlauftemperaturen der Wärmeerzeugungsanlagen, während Ausgangsdaten wie minimale und maximale Drücke oder Temperaturen in der späteren Optimierung als Restriktionen verwendet werden. Die Ein- und Ausgangsdaten werden zunächst gesammelt, aufbereitet und anschließend für das Training von Modellen der Systemidentifikation oder des maschinellen Lernens verwendet. Normalerweise kommen Messdaten für die Ein- und Ausgänge zum Einsatz. Hier werden in beiden Fällen die Modelle in *SIR 3S*[®] für die Datenerzeugung verwendet. Somit müssen zunächst keine Messdaten gesammelt werden und die vereinfachten Modelle können basierend auf komplexeren physikalischen Simulationen erzeugt werden. Auch zukünftige Ausbauzustände können so berücksichtigt werden. Die *SIR 3S*[®] Modelle werden für Methoden des maschinellen Lernens hydraulisch und thermisch stationär und für die Systemidentifikation thermisch instationär verwendet. Die verschiedenen Algorithmen der beiden Methoden wurden bereits in den vorab genannten Kapiteln beschrieben. Nun werden zunächst die Parameter und Ergebnisse vereinfachter stationärer Modelle untersucht.

Datenerzeugung

Für das Training mit Algorithmen des maschinellen Lernens werden 144 260 stationäre Simulationen mit variierenden Eingangsgrößen mit dem aggregierten und vollständigen Netzmodell durchgeführt. Fehlerhafte Simulationen werden vorab durch die Software *SIR 3S*[®] identifiziert oder im Nachhinein gelöscht. Dazu zählen Simulationen mit Differenz-, Vor- oder Rücklaufdrücken kleiner 0 bar und Simulationen mit großer Abweichung zwischen dem vorgegebenen und simulierten Wärmeverbrauch. Auch Ergebnisse mit Massenströmen außerhalb des Betriebsbereichs der Umwälzpumpen werden nicht berücksichtigt. Für die Erzeugung der Trainingsdaten werden die Eingangsgrößen Wärmeabnahme, Vorlauftemperatur und Vorlaufdruck jeweils zwischen 0 und 100 % variiert. Die Differenzdruckvorgabe an den Schlechtpunkten wird konstant entsprechend der zur Verfügung gestellten Werte im Leitsystem eingestellt. Die DEA wird nicht verwendet, da sie an den vier als Referenz verwendeten Beispieltagen nicht zum Einsatz kam. Die Rücklauftemperatur am HKW wird wie für die Modellvalidierung in *SIR 3S*[®] konstant belassen. Die Eingangsgrößen bilden bis auf den Einsatz der DEA und die konstanten Rücklauftemperaturen den gesamten Betriebsbereich des Fernwärmenetzes ab.

Modelle werden mit den folgenden elf für die Optimierung benötigten Ausgangsgrößen trainiert, welche in der Optimierung als Zielgrößen oder Restriktionen eingesetzt werden: Wärmeverlust, Pumpendrehzahl und -leistung, Umwälzmassenstrom, minimale und maximale Strömungsgeschwindigkeit, minimale Vorlauftemperatur, minimaler Differenzdruck, Rücklaufdruck sowie minimale und maximale Drücke. Insgesamt soll durch die Modelle also der Zusammenhang zwischen drei Eingangs- und elf Ausgangsgrößen erlernt werden. Dafür stehen 144 260 Datensätze aus Ein- und Ausgangsgrößen zur Verfügung.

Vergleich verschiedener Ansätze des maschinellen Lernens

Für die Vereinfachung stationärer Modelle von Fernwärmenetz 1 werden einige der in Kapitel 2.5 vorgestellten Ansätze des maschinellen Lernens in *Python* 3.8.12 verglichen:

- Lineare Regression mit der Bibliothek *scikit-learn* [98] in Version 1.0.2
- Polynomregression (*scikit-learn* in Version 1.0.2)
- *Random Forests* (*scikit-learn* in Version 1.0.2)
- XGBoost in der Bibliotheksversion 1.5.2
- Neuronale Netze in den Bibliotheken *TensorFlow/Keras* [2] in Version 2.3.0 und *PyTorch* [97] in Version 1.11.0

Die Ansätze werden mit zunehmender Komplexität von einfachen, statistischen Modellen wie der linearen Regression und Polynomregression bis hin zu fortgeschrittenen Algorithmen untersucht. Dabei sollen die Ansätze des *Ensemble Learnings* XGBoost und *Random Forests* sowie neuronale Netze angewandt werden, da diese besonders gute Ergebnisse bezüglich Lernproblemen mit strukturierten Daten zeigen [42, S. 28].

Die Datensätze werden zufällig in 80 % Trainingsdaten (115 408 Datenätze) und 20 % Testdaten (28 852 Datensätze) geteilt. 80 % der Daten werden also verwendet, um ein Modell zu trainieren und 20 %, um es anschließend zu bewerten. Für neuronale Netze werden zusätzlich 20 % der Daten für die Validierung von Parametern während des Trainings verwendet und somit nur 60 % (86 556 Datenpunkte) für das Training. Alle Eingangs- und Ausgangsdaten werden zwischen 0 und 1 normiert. Die Auswertung der Modelle erfolgt auf Basis der folgenden Bewertungskriterien der *Python* Bibliothek *scikit-learn* [98], welche jeweils von den n vom Modell vorhergesagten Ausgangswerten $\hat{\mathbf{y}}$ und den n tatsächlichen Simulationswerten \mathbf{y} aus *SIR 3S*[®] abhängen:

$$MAPE(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{n} \cdot \sum_{i=0}^{n-1} \frac{|y_i - \hat{y}_i|}{\max(\epsilon, |\hat{y}_i|)} \quad (6.1)$$

$$MAE(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{n} \cdot \sum_{i=0}^{n-1} |y_i - \hat{y}_i| \quad (6.2)$$

$$MSE(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{n} \cdot \sum_{i=0}^{n-1} (y_i - \hat{y}_i)^2 \quad (6.3)$$

$$RMSE(\mathbf{y}, \hat{\mathbf{y}}) = \sqrt{\frac{1}{n} \cdot \sum_{i=0}^{n-1} (y_i - \hat{y}_i)^2} \quad (6.4)$$

$$R^2(\mathbf{y}, \hat{\mathbf{y}}) = 1 - \frac{\sum_{i=0}^{n-1} (y_i - \hat{y}_i)^2}{\sum_{i=0}^{n-1} (y_i - \bar{y})^2} \quad (6.5)$$

Die Größe MAPE beschreibt den *Mean Absolute Percentage Error*, also den durchschnittlichen absoluten prozentualen Fehler. Der *Mean Absolute Error* (MAE) ist der durchschnittliche absolute Fehler, der *Mean Squared Error* (MSE) der durchschnittliche quadratische Fehler, der *Root Mean Squared Error* (RMSE) die Wurzel des mittleren quadratischen Fehlers und der R^2 -Wert ist das Bestimmtheitsmaß. Das Bestimmtheitsmaß gibt den Anteil der Varianz an, welcher von dem Modell beschrieben werden kann und wird als Beurteilungskriterium für Regressionen verwendet. Für die Fehlerwerte gilt, dass niedrigere Werte besser sind. Der R^2 -Wert kann maximal einen Wert von 1 annehmen. Höhere Werte sind hier besser.

Tabelle 6.1 zeigt die Auswertung verschiedener Modelle, welche jeweils mit den Trainingsdaten trainiert und anschließend mit den Testdaten bewertet werden. In der Tabelle sind zunächst die durchschnittlichen relativen und absoluten Abweichungen für den minimalen Differenzdruck Δp gegeben. Diese

Größen zeigen im Schnitt die größten Abweichungen der Modelle. Zusätzlich werden der MAPE und R^2 -Wert im Durchschnitt über alle elf Ausgänge berechnet sowie die Rechenzeit (t_s) für stationäre 24-Stunden-Simulationen im 15-Minuten-Takt aus jeweils fünf Rechnungen bestimmt. Die Simulationen werden auf einem Rechner mit *Intel Core i7-4771* (4 x 3,50GHz) und 32 GB RAM durchgeführt. Für neuronale Netze wird jeweils die Anzahl der Schichten s und der Neuronen n angegeben.

Tabelle 6.1.: Auswertung verschiedener Machine Learning Modelle mit Testdaten für Netz 1.

Modell	MAPE Δp (%)	MAE Δp (bar)	MAPE gesamt (%)	R^2 gesamt	t_s (ms)
Lineare Regression	71,54	0,193	13,73	0,9337	0,4
Polynomregression ($n = 2$)	19,44	0,088	3,85	0,99022	0,55
Polynomregression ($n = 5$)	17,52	0,053	1,97	0,99713	0,7
Polynomregression ($n = 10$)	8,195	0,027	0,96	0,9984	1
Polynomregression ($n = 20$)	5,965	0,0084	0,71	0,99855	2,6
<i>Random Forests</i>	2,65	0,011	0,31	0,99756	23
XGBoost	3,278	0,014	0,537	0,99853	99
Neuronales Netz ($s = 1, n = 22$)	20,266	0,077	2,726	0,99448	2,3
Neuronales Netz ($s = 2, n = 66/22$)	6,639	0,021	0,932	0,99702	2,8
Neuronales Netz ($s = 3, n = 88/44/22$)	5,429	0,013	0,714	0,99764	3,2
Neuronales Netz ($s = 4, n = 66/44/44/11$)	6,298	0,015	0,811	0,99774	3,6

Die Auswertung zeigt, dass die lineare Regression nicht geeignet ist, um Netz 1 über alle Betriebsbereiche abzubilden. Da nur lineare Funktionen trainiert werden, war dieses Ergebnis zu erwarten. Theoretisch kann lineare Regression in mehreren Betriebsbereichen durchgeführt werden, um so mehrere Modelle zu erhalten, die dann miteinander verknüpft werden müssen. Dieser Ansatz wurde noch nicht untersucht. Bei der Polynomregression steigt die Genauigkeit der Modelle mit zunehmendem Grad des Polynoms. Damit geht eine Zunahme der Simulationszeit und des Arbeitsspeicherbedarfs einher. Die besten Ergebnisse zeigen die Modelle *Random Forests* und XGBoost, welche beide auf Entscheidungsbäumen basieren und in Kapitel 2.5.5 zum *Ensemble Learning* erläutert wurden. Auffällig ist, dass die beiden Algorithmen im Vergleich deutlich länger für eine Simulation benötigen. Die Parameter der Algorithmen wurden dabei in den Standardeinstellungen belassen. Bei dem *Random Forests* Algorithmus werden so zum Beispiel 100 parallele Bäume trainiert, was zu einem hohen Festplattenspeicherbedarf von 1,95 GB führt.

Für neuronale Netze werden verschiedene Konfigurationen mit mehreren Schichten und Anzahlen der Neuronen in der Bibliothek *TensorFlow / Keras* getestet. Dabei werden Netze mit ein bis vier Schichten und verschiedenen Anzahlen von Neuronen (Multiple der Anzahl der 11 Ausgänge) untersucht. Die Modelle werden jeweils für 100 Epochen mit dem Optimierer *Adam*, dem MAE als *Loss*-Funktion und der Aktivierungsfunktion *Swish* trainiert. Die Gewichte werden mit der Funktion *He-Uniform*, auch *Kaiming-Uniform* genannt, initialisiert. Diese Funktionen werden zunächst als Test für alle neuronalen Netze verwendet, um untereinander die Anzahl der Schichten und Neuronen zu vergleichen. Im späteren Verlauf dieser Arbeit werden genauere Untersuchungen bezüglich der Parameter für neuronale Netze durchgeführt. Die Ergebnisse für vier der Modelle sind in Tabelle 6.1 dargestellt.

Im nächsten Schritt werden die Algorithmen auf vier Beispieltage angewandt, um eine Übersicht über die Modellgenauigkeit für unterschiedliche Betriebsbereiche des Fernwärmenetzes zu erhalten. Die Auswertung ist in Tabelle 6.2 für den Wintertag 1 und in Tabelle 6.3 für den Sommertag 3

dargestellt. Hier wird für den minimalen Differenzdruck Δp der maximale absolute Fehler statt dem *Mean Absolute Error* (MAE) angegeben, um zusätzlich maximale Rechenfehler abschätzen zu können. Auch der maximale relative Fehler über alle Ausgänge wird deshalb festgehalten.

Tabelle 6.2.: Abweichung verschiedener Machine Learning Modelle zu *SIR 3S*[®] an Tag 1.

Modell	<i>MAPE</i> Δp (%)	Max. abs. Δp (bar)	<i>MAPE</i> gesamt (%)	Max. rel. Error (%)
Lineare Regression	14,92	0,24	4,64	35,92
Polynomregression ($n = 10$)	1,65	0,032	0,31	2,77
Polynomregression ($n = 20$)	1,06	0,017	0,24	2,23
<i>Random Forests</i>	2,57	0,047	1,02	7,09
XGBoost	5,47	0,099	1,82	16,24
Neuronales Netz ($s = 3, n = 88/44/22$)	0,56	0,013	0,22	1,29

Tabelle 6.3.: Abweichung verschiedener Machine Learning Modelle zu *SIR 3S*[®] an Tag 3.

Modell	<i>MAPE</i> Δp (%)	Max. abs. Δp (bar)	<i>MAPE</i> gesamt (%)	Max. rel. Error (%)
Lineare Regression	12,74	0,46	27,77	101,697
Polynomregression ($n = 10$)	22,69	1,6	4,91	47,55
Polynomregression ($n = 20$)	212	29,42	38,42	873,573
<i>Random Forests</i>	1,23	0,086	5,42	41,95
XGBoost	1,39	0,154	4,82	42,3
Neuronales Netz ($s = 3, n = 88/44/22$)	0,50	0,044	1,76	18,1

Für alle Algorithmen fällt auf, dass der durchschnittliche absolute prozentuale Fehler (MAPE) für den Wintertag 1 kleiner ist als für die Testdaten in Tabelle 6.1 und für den Sommertag 3 größer. Ein Vergleich der beiden Tage untereinander zeigt sowohl absolut als auch relativ höhere Abweichungen an dem Sommertag. Mit linearer Regression lassen sich wie mit den Testdaten keine guten Ergebnisse erzeugen. Die Modelle der Polynomregression mit hohem Polynomgrad zeigen für den Wintertag gute Ergebnisse, am Sommertag treten allerdings hohe Abweichungen auf. Zudem erzeugen nun die Algorithmen basierend auf Entscheidungsbäumen schlechtere Ergebnisse. Ein Grund kann der Aufbau der Eingangsdaten sein, welche in diskreten Schritten vom Minimum bis zum Maximum des Betriebsbereichs gewählt wurden. Baumalgorithmen können so sehr gut die Ergebnisse für die jeweiligen Schritte lernen. Werden anschließend tatsächliche Eingangsdaten verwendet, welche zwischen den diskreten Schritten liegen, weichen die Ergebnisse stärker ab. Die Baumalgorithmen sind also an die Lerndaten überangepasst. Behoben werden könnte dieses Problem durch Verwendung anderer Parameter für die Algorithmen, was aber in einer weiteren Studie zu testen wäre.

Für beide Beispieltage liefert das neuronale Netz die besten Ergebnisse. Mit neuronalen Netzen ist es insgesamt möglich, mit geringen Veränderungen der Parameter sehr gute Ergebnisse zu erzielen. Auch die Simulationszeiten der Modelle sind vergleichsweise niedrig. Aus diesen Gründen sollen neuronale Netze für weitere detaillierte Untersuchungen verwendet werden. Dabei werden die Parameter und Architekturen der Netze in den folgenden Abschnitten so gewählt, dass stationäre Simulationen in *SIR 3S*[®] möglichst genau abgebildet, während Rechenzeiten deutlich reduziert werden. Dafür wird hier zu der Bibliothek *PyTorch* gewechselt, welche mittlerweile häufiger verwendet wird als *TensorFlow* oder *TensorFlow/Keras* [57]. Zusätzlich werden die *PyTorch* Modelle nach dem Training in das *Open Neural Network Exchange* (ONNX) Format konvertiert und mit der *ONNX-Runtime* ausgeführt. So ist eine weitere Reduzierung der Simulationszeiten der neuronalen Netze möglich.

Überanpassung

Zunächst soll untersucht werden, wo die Grenzen der neuronalen Netze liegen und ab wann es zu einer Überanpassung (*Overfitting*) an die Trainingsdaten kommt. Dies kann besonders bei großen Netzen, der Verwendung des MSE als *Loss*-Funktion und einer geringen Anzahl an nicht zufällig gewählten Trainingsdaten beobachtet werden. Abbildung 6.3 zeigt beispielhaft das Training eines neuronalen Netzes mit einer versteckten Schicht mit einer Anzahl von 1000 Neuronen sowie einer zunehmenden Anzahl an verwendeten Datensätzen. In der ersten Abbildung 6.3a werden nur 1470 der 115 408 Datensätze für das Training (*Trainingsloss* in Blau) verwendet und die restlichen Daten für die Validierung (*Validierungsloss* in Orange). Hier ist ersichtlich, dass der *Trainingsloss* nahezu bei null, während der *Validierungsloss* deutlich höher liegt und im Laufe des Trainings ansteigt. In den folgenden drei Abbildungen 6.3b bis 6.3d wird die Anzahl der Datensätze für das Training auf 14 705, 29 410 und 73 526 erhöht. Zuletzt kommt es offensichtlich nicht mehr zu einer Überanpassung. Wird statt des MSE, der MAE als *Loss*-Funktion verwendet, kommt es auch bei einer geringen Anzahl an Trainingsdaten kaum zu anhand der Validierungsdaten erkennbarer Überanpassung. Für das Training der neuronalen Netze mit zufällig gewählten 60 % der Daten, also 86 556 Datensätzen, stellt *Overfitting* so vermutlich nur ein geringes Problem dar.

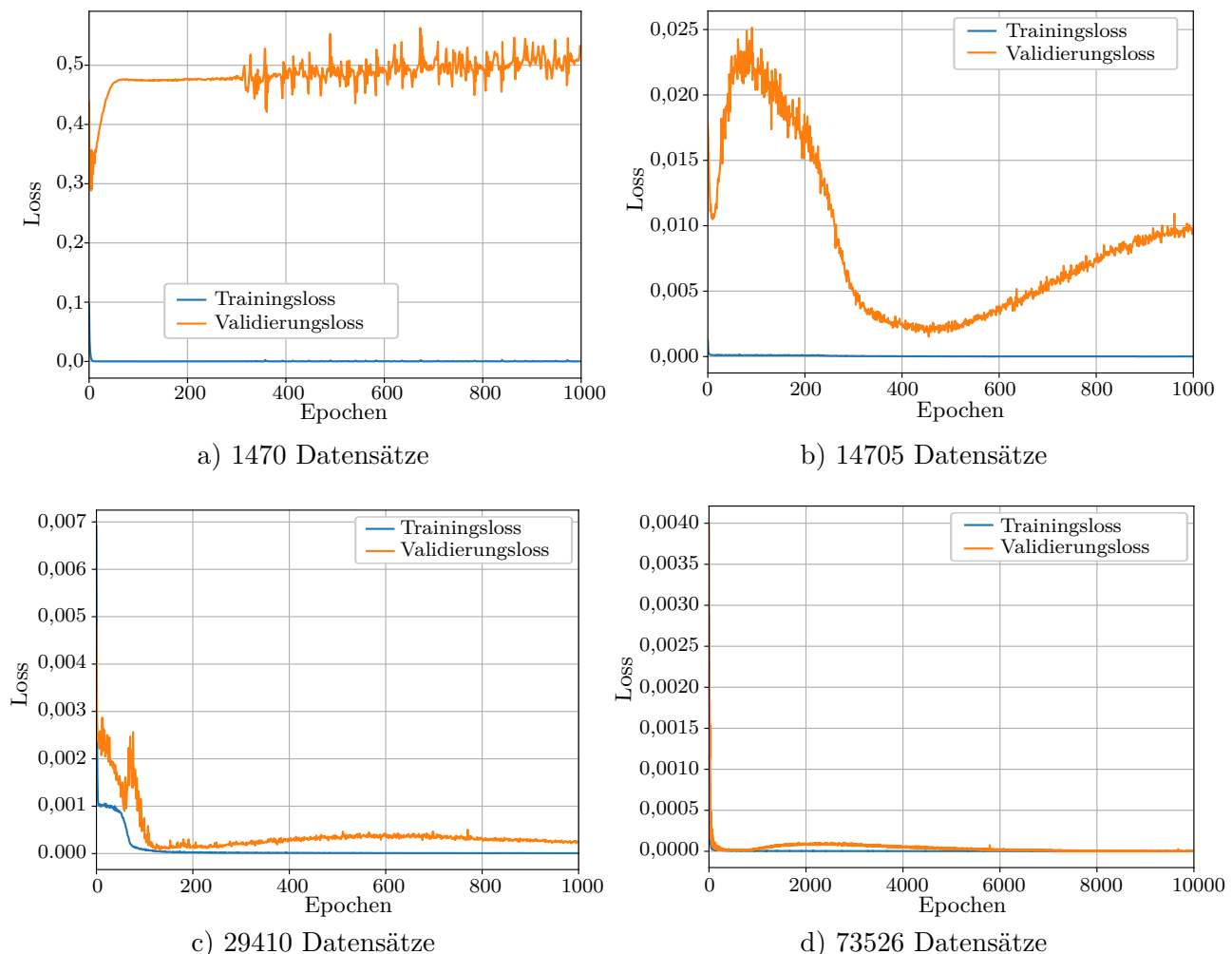


Abbildung 6.3.: Beispiel für *Overfitting* eines neuronalen Netzes für Fernwärmenetz 1 mit unterschiedlicher Anzahl der für das Training verwendeten Datensätze.

Die Problematik sollte für ein trainiertes Modell dennoch mit Vergleichssimulationen einiger Beispieltage zu unterschiedlichen Jahreszeiten untersucht werden. Wintertage zeigen aufgrund höherer Wärmeverbräuche verglichen mit Sommertagen größere Werte bei relevanten Ausgangsgrößen wie Leistungen, Massenströmen und Strömungsgeschwindigkeiten. Entsprechend sind auch absolute Abweichungen bei relativ ähnlichen Fehlern größer. Beim Training der neuronalen Netze, vor allem unter Verwendung des mittleren quadratischen Fehlers (MSE) als *loss*-Funktion, kann es daher zu einer Überanpassung an Wintertage mit größeren Werten kommen, was für Sommertage später zu einer geringeren Genauigkeit führt. Dieser Effekt ist in der Genauigkeit der Validierungsdaten nicht immer ersichtlich. Dennoch zeigt die Untersuchung, dass nicht unbedingt die volle Anzahl der erzeugten Simulationsdatensätze für das Training von neuronalen Netzen mit guter Genauigkeit notwendig ist. Hier könnte in einem weiteren Schritt im Detail untersucht werden, wie viele Daten tatsächlich für das Training erforderlich sind, um so potenziell die Simulationszeit für die Datenerzeugung zu reduzieren.

Untersuchung der Anzahl der Schichten und Neuronen

Nun wird erneut eine Untersuchung der Anzahl der Schichten und Neuronen pro Schicht durchgeführt, um eine möglichst gute Abbildung von Fernwärmenetz 1 durch ein neuronales Netz zu erreichen. Folgende Einstellungen werden in *PyTorch* als Basis für diese Untersuchung verwendet:

- Optimierer: *Adam* mit einer Lernrate von 10^{-3}
- *Loss*-Funktion: MAE
- Aktivierungsfunktion: *Swish*, bzw. *SiLU* in *PyTorch*
- Initialisierung der Gewichte mit der Voreinstellung in *PyTorch*: *He*-, bzw. *Kaiming-Uniform*
- 60%/20%/20% Unterteilung und Normierung der Ein- und Ausgangsdaten zwischen 0 und 1

Der Programmcode für die Klasse neuronaler Netze in *PyTorch* und deren Training befindet sich im Anhang A.1.1. Tabelle A.1 zeigt dort eine Auswertung neuronaler Netze mit ein bis vier Schichten nach einem Training für 100 Epochen, Tabelle A.3 zusätzlich einige Netze, für welche das Training auf 1000 Epochen ausgeweitet wurde. Für die Bewertung der Netze wird im Anhang eine *Score* festgelegt, entsprechend welcher jeweils die besten Netze mit ein bis vier Schichten ausgewählt und für weitere Untersuchungen verwendet werden.

Untersuchung der *Loss*-Funktion und Lernrate

Des Weiteren werden Modelle, die mit dem mittleren quadratischen Fehler MSE und dem mittleren absoluten Fehler MAE als *loss*-Funktion trainiert werden, verglichen. Zusätzlich werden die Modelle einmal für 1000 Epochen mit einer konstanten Lernrate von 10^{-3} sowie mit einer alle 200 Epochen halbierten Lernrate trainiert. Dieses Vorgehen wird auch *Learning-Rate-Decay* genannt. Tabelle 6.4 zeigt einige der Bewertungskriterien für diese neuronalen Netze. Die in der Tabelle mit t_s bezeichnete durchschnittliche Rechenzeit der Modelle wird aus 100 in Schleife durchgeführten 24-Stunden-Simulationen im 15-Minuten-Takt berechnet.

Tabelle 6.4.: Bewertung neuronaler Netze mit Testdaten für Fernwärmenetz 1.

Neuronen pro Schicht (<i>Loss</i> -Funktion)	<i>MAPE</i> Δp (%)	<i>MAE</i> Δp (bar)	<i>MAPE</i> gesamt	R^2	t_s (ms)
176 (MSE)	17,98	0,031	1,952	0,99797	0,060
66/88 (MSE)	6,23	0,016	0,845	0,99860	0,067
132/44 (MSE)	5,12	0,017	0,680	0,99853	0,076
110/88/44 (MSE)	4,48	0,013	0,602	0,99879	0,096

Tabelle 6.4 fortgesetzt.

Neuronen pro Schicht (<i>Loss</i> -Funktion)	<i>MAPE</i> Δp (%)	<i>MAE</i> Δp (bar)	<i>MAPE</i> gesamt	R^2	t_s (ms)
176/44/88 (MSE)	6,90	0,019	0,821	0,99869	0,102
110/44/44/22 (MSE)	6,78	0,016	0,850	0,99887	0,121
176 (MAE)	109,22	0,028	10,224	0,99685	0,061
66/88 (MAE)	12,09	0,014	1,295	0,99732	0,064
132/44 (MAE)	9,29	0,012	1,036	0,99730	0,073
110/88/44 (MAE)	5,23	0,013	0,650	0,99742	0,098
176/44/88 (MAE)	5,93	0,011	0,709	0,99817	0,110
110/44/44/22 (MAE)	4,56	0,011	0,586	0,99802	0,119
176 (MAE) ohne <i>Decay</i>	14,03	0,025	1,630	0,99670	0,064
66/88 (MAE) ohne <i>Decay</i>	6,26	0,012	0,816	0,99789	0,073
132/44 (MAE) ohne <i>Decay</i>	7,42	0,012	0,894	0,99805	0,075
110/88/44 (MAE) ohne <i>Decay</i>	5,19	0,013	0,732	0,99787	0,096
176/44/88 (MAE) ohne <i>Decay</i>	6,22	0,011	0,910	0,99810	0,103
110/44/44/22 (MAE) ohne <i>Decay</i>	4,28	0,011	0,582	0,99822	0,120

Insgesamt ergeben sich sowohl in den Tabellen im Anhang als auch in Tabelle 6.4 nur geringe Unterschiede in den Genauigkeiten der trainierten Modelle. Auch die Simulationszeiten sind ähnlich und im Bereich von 0,1 ms. Am auffälligsten ist der Unterschied des mittleren absoluten prozentualen Fehlers MAPE für den minimalen Differenzdruck Δp . Aber auch hier unterscheidet sich der mittlere absolute Fehler MAE nur wenig. Der MAPE im Durchschnitt über alle Ausgänge ist für die Mehrzahl der Modelle kleiner 1 %. Alle Modelle mit einem höheren MAPE sowie einem MAPE des Differenzdrucks größer 6 % sind in der Tabelle markiert und werden nicht weiter untersucht. Für eine Entscheidung zwischen den restlichen Modellen werden die neuronalen Netze wieder auf die vier Beispieltage angewandt. Die Auswertung ist in Tabelle 6.5 für Tag 1 und in Tabelle 6.6 für Tag 3 dargestellt.

Tabelle 6.5.: Abweichung verschiedener neuronaler Netze zu *SIR 3S*[®] an Tag 1.

Neuronen pro Schicht (<i>Loss</i> -Funktion)	<i>MAPE</i> Δp (%)	Max. abs. Δp (bar)	<i>MAPE</i> gesamt (%)	Max. rel. Error (%)
132/44 (MSE)	1,56	0,021	0,313	3,83
110/88/44 (MSE)	1,46	0,018	0,235	1,46
110/88/44 (MAE)	0,49	0,010	0,190	0,96
176/44/88 (MAE)	0,51	0,012	0,198	1,37
110/44/44/22 (MAE)	0,55	0,012	0,202	1,10
110/88/44 (MAE) ohne <i>Decay</i>	0,54	0,014	0,346	1,32
110/44/44/22 (MAE) ohne <i>Decay</i>	0,46	0,011	0,199	0,92

Tabelle 6.6.: Abweichung verschiedener neuronaler Netze zu *SIR 3S*[®] an Tag 3.

Neuronen pro Schicht (<i>Loss</i> -Funktion)	<i>MAPE</i> Δp (%)	Max. abs. Δp (bar)	<i>MAPE</i> gesamt (%)	Max. rel. Error (%)
132/44 (MSE)	20,72	1,672	5,145	104,58
110/88/44 (MSE)	60,63	3,979	11,543	309,94
110/88/44 (MAE)	1,93	0,307	2,494	31,07
176/44/88 (MAE)	1,04	0,206	1,714	31,73
110/44/44/22 (MAE)	0,41	0,049	1,163	10,10
110/88/44 (MAE) ohne <i>Decay</i>	0,17	0,023	0,791	9,21
110/44/44/22 (MAE) ohne <i>Decay</i>	0,52	0,042	1,785	13,05

Alle Modelle zeigen für Beispieltag 1 im Winter mit hohem Wärmeverbrauch sehr gute Ergebnisse und durchschnittliche relative Fehler unter 0,5 %. Für den zweiten Beispieltag im Sommer mit niedrigem Wärmeverbrauch sind die Abweichungen zu *SIR 3S*[®] größtenteils ebenfalls gering. Jedoch führen die Modelle, die mit dem mittleren quadratischen Fehler MSE trainiert wurden, teils zu großen Abweichungen trotz sehr guter Ergebnisse über die gesamten Testdaten nach Tabelle 6.4. Die Modelle sind vermutlich an Tage mit hohem Wärmeverbrauch überangepasst. An Tagen mit geringem Wärmeverbrauch sind entsprechend auch viele der Zielgrößen des Trainings klein. Die Abweichungen an Tagen mit hohem Wärmeverbrauch und großen Zielgrößen gehen so durch das Quadrieren der Fehler stärker in die Zielfunktion des Trainings ein. Diese Beobachtung bestätigt die zuvor durchgeführte Untersuchung bezüglich *Overfitting* der Modelle. Somit wird für folgende Analysen nur noch der MAE als *Loss*-Funktion verwendet.

Auch die mit dem mittleren absoluten Fehler trainierten neuronalen Netze weisen hohe maximale relative Abweichungen auf, welche hauptsächlich für die Berechnung der minimalen Vorlauftemperatur auftreten. Es ist auffällig, dass die Modelle mit abnehmender Lernrate für den Sommertag größtenteils schlechtere Ergebnisse und höhere Abweichungen zu *SIR 3S*[®] zeigen als die neuronalen Netze mit konstanter Lernrate. Für den Wintertag sind die Ergebnisse hingegen besser. Auch hier sind die neuronalen Netze vermutlich an die Trainingsdaten der Wintertage überangepasst und Vorhersagen für Tage mit niedrigen Wärmeverbräuchen schlechter. Abschließend wird ein neuronales Netz mit dem MAE als *Loss*-Funktion und konstanter Lernrate empfohlen. Aufgrund der insgesamt sehr guten Ergebnisse sowohl für den Sommer- als auch für den Wintertag wird das Modell mit drei Schichten sowie 110, 88 und 44 Neuronen ausgewählt.

Auswahl der Aktivierungsfunktion und Initialisierung der Gewichte

Bisher wird für das Training der neuronalen Netze die Aktivierungsfunktion *Swish* für alle versteckten Schichten verwendet. Neben *Swish* werden nach Kapitel 2.5.4 die Aktivierungsfunktionen Tangens Hyperbolicus, die Sigmoid Funktion und die Funktion ReLU getestet. Für *Swish* und ReLU werden die Gewichte mit der für ReLU entwickelten Funktion *He*- bzw. *Kaiming-Uniform* initialisiert. Zusätzlich wird die Funktion *He*- bzw. *Kaiming-Normal* für beide Aktivierungsfunktionen getestet. Für Tangens Hyperbolicus und die Sigmoid Funktion wird mit *Xavier*- bzw. *Glorot-Uniform* initialisiert. Den Vergleich der Netze mit *SIR 3S*[®] zeigt Tabelle 6.7.

Tabelle 6.7.: Bewertung des neuronalen Netzes mit 110, 88 und 44 Neuronen basierend auf Testdaten mit verschiedenen Aktivierungsfunktionen für Fernwärmenetz 1.

Aktivierungsfunktion	<i>MAPE</i> Δp (%)	<i>MAE</i> Δp (bar)	<i>MAPE</i> gesamt	R^2
<i>Swish He-Uniform</i>	5,19	0,013	0,732	0,99787
<i>Swish He-Normal</i>	6,32	0,011	0,761	0,99797
ReLU <i>He-Uniform</i>	6,02	0,010	0,707	0,99806
ReLU <i>He-Normal</i>	6,36	0,012	0,763	0,99789
Sigmoid	9,14	0,013	1,224	0,99746
Tangens Hyperbolicus	6,16	0,014	0,966	0,99752

Die mit *Swish* und ReLU trainierten neuronalen Netze unterscheiden sich in den Ergebnissen kaum. Die Aktivierungsfunktionen Sigmoid und Tangens Hyperbolicus führen zu etwas schlechteren Ergebnissen. Tabelle 6.8 zeigt einen weiteren Vergleich der neuronalen Netze für den Winter- und Sommerbeispieltag. An dem Wintertag zeigt das Modell mit der Aktivierungsfunktion *Swish* und Initialisierung der Gewichte mit *He-Normal* noch etwas bessere Ergebnisse als das bisher verwendete neuronale Netz mit *He-Uniform*. Die mit ReLU trainierten Netze zeigen ähnlich gute Ergebnisse, allerdings höhere relative Abweichungen. Für den Sommertag liefert die Kombination von *Swish* und Initialisierung der Gewichte mit *He-Uniform* die geringsten Abweichungen. Insgesamt wird dieser Kombination aus Aktivierungsfunktion und Initialisierung beibehalten.

Tabelle 6.8.: Abweichung des neuronalen Netzes mit 110, 88 und 44 Neuronen sowie verschiedenen Aktivierungsfunktionen zu *SIR 3S*[®] an Tag 1 im Januar und 3 im Juli.

Aktivierungsfunktion	<i>MAPE</i> Δp (%)	Max. abs. Δp (bar)	<i>MAPE</i> gesamt (%)	Max. rel. Error (%)
<i>Swish He-Uniform</i> , Tag 1	0,54	0,014	0,346	1,32
<i>Swish He-Normal</i> , Tag 1	0,55	0,010	0,189	1,11
ReLU <i>He-Uniform</i> , Tag 1	1,39	0,023	0,369	4,27
ReLU <i>He-Normal</i> , Tag 1	0,80	0,013	0,344	2,32
<i>Swish He-Uniform</i> , Tag 3	0,17	0,023	0,791	9,21
<i>Swish He-Normal</i> , Tag 3	0,22	0,015	1,239	11,39
ReLU <i>He-Uniform</i> , Tag 3	0,39	0,031	0,957	10,88
ReLU <i>He-Normal</i> , Tag 3	0,26	0,024	1,124	10,18

Normierung vs. Standardisierung

Weiterhin wird die Normierung und Standardisierung der Daten verglichen. Dafür wird das neuronale Netz wie bisher mit Normierung zwischen 0 und 1 trainiert sowie zusätzlich mit Normierung der Ein- und Ausgangsdaten zwischen -1 und 1 und mit Standardisierung, bei welcher die Werte so verteilt werden, dass das arithmetische Mittel der Daten 0 und deren Varianz 1 ist [42, S. 278]. Die folgende Tabelle 6.9 zeigt den Vergleich der resultierenden neuronalen Netze für Beispieltag 1 und 3.

Tabelle 6.9.: Abweichung des neuronalen Netzes mit 110, 88 und 44 Neuronen mit unterschiedlicher Datenskalierung zu *SIR 3S*[®] an Tag 1 im Januar und 3 im Juli.

Art der Skalierung	<i>MAPE</i> Δp (%)	Max. abs. Δp (bar)	<i>MAPE</i> gesamt (%)	Max. rel. Error (%)
Normierung 0 bis 1, Tag 1	0,54	0,014	0,346	1,32
Normierung -1 bis 1, Tag 1	0,46	0,010	0,312	1,27
Standardisierung, Tag 1	0,38	0,010	0,157	0,87

Tabelle 6.9 fortgesetzt

Art der Skalierung	<i>MAPE</i> Δp (%)	Max. abs. Δp (bar)	<i>MAPE</i> gesamt (%)	Max. rel. Error (%)
Normierung 0 bis 1, Tag 3	0,17	0,023	0,791	9,21
Normierung -1 bis 1, Tag 3	0,63	0,027	0,878	17,606
Standardisierung, Tag 3	0,18	0,012	0,939	8,72

Die Normierung zwischen -1 und 1 liefert für Tag 1 leicht verbesserte Ergebnisse verglichen mit der Normierung zwischen 0 und 1. Allerdings sinkt die Genauigkeit für Tag 3. Die Skalierung kann die Genauigkeit an Tag 1 verbessern. Für Tag 3 ist das Ergebnis nicht eindeutig. Hier nimmt mit der Skalierung der MAPE über alle Ausgangsgrößen verglichen zur Normierung zwischen 0 und 1, leicht ab. Da besonders warme Tage mit niedrigem Wärmeverbrauch wie Tag 3 in den bisherigen und folgenden Untersuchungen schlechte Ergebnisse zeigen, wird weiterhin die Normierung zwischen 0 und 1 verwendet.

Nicht untersuchte Parameter und Optionen

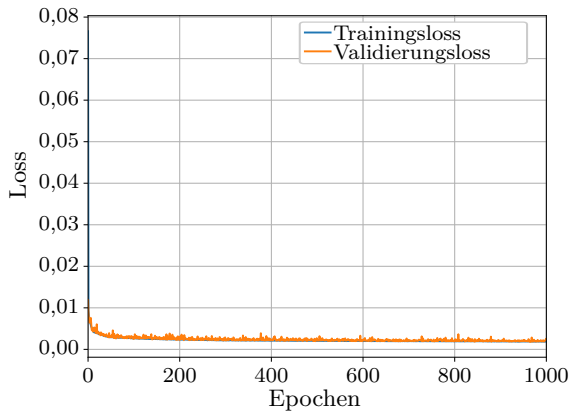
Einige Parameter der neuronalen Netze in *PyTorch* wurden bisher nicht untersucht. Dazu gehört das Training der Netze mit weiteren Optimierern. Auch eine Regularisierung der Gewichte oder *Early-Stopping* des Trainings wurden nicht analysiert. Der *Dropout* einiger Neuronen hingegen wurde getestet. In diesem Fall wird in jeder versteckten Schicht eine vorgegebene Prozentzahl der Neuronen deaktiviert und nicht für das Training verwendet. Eine Verbesserung der Trainingsergebnisse konnte dabei bisher nicht erreicht werden. Mit den Ansätzen der Regularisierung, des *Early-Stopplings* und des *Dropouts* einiger Neuronen kann die Überanpassung der neuronalen Netze an Trainingsdaten verhindert werden. Eine Überanpassung wird hauptsächlich bei Verwendung des MSE als *Loss*-Funktion beobachtet. Allerdings zeigen auch mit abnehmender Lernrate trainierte neuronale Netze zwar über alle Testdaten gute Ergebnisse, weisen jedoch für Sommertage mit niedrigen Verbräuchen höhere Abweichungen als für Wintertage auf. Aufgrund der sehr guten Ergebnisse einiger trainierter neuronaler Netze werden weitere Parameter hier nicht untersucht, kommen aber infrage, falls Verbesserungen der Modelle notwendig werden.

Neuronale Netze für Fernwärmenetz 1

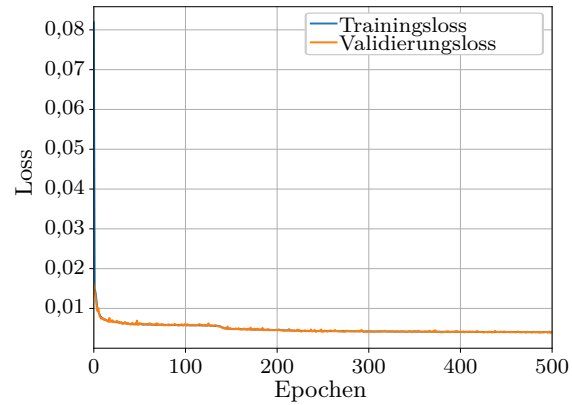
Zusammenfassend werden folgende Einstellungen für das neuronale Netz des aggregierten Fernwärmenetzes 1 gewählt:

- Drei verborgene Schichten mit 110, 88 und 44 Neuronen
- Optimierer: *Adam* mit einer Lernrate von 10^{-3} und Training für 1000 Epochen
- *Loss*-Funktion: MAE
- Aktivierungsfunktion: *Swish* und Initialisierung der Gewichte mit *He-Uniform*
- 60%/20%/20% Unterteilung und Normierung der Ein- und Ausgangsdaten zwischen 0 und 1

Diese Konfiguration liefert insgesamt gute Ergebnisse im Vergleich mit *SIR 3S*[®] und einen Kompromiss zwischen der Genauigkeit für Tage mit hohem und niedrigem Fernwärmeverbrauch. Auch für das vollständige Fernwärmenetz 1 wird ein neuronales Netz in dieser Konfiguration gewählt und jeweils für 100, 500 und 1000 Epochen mit und ohne abnehmende Lernrate trainiert. Das Training mit 500 Epochen und konstanter Lernrate ergibt das beste Modell. Die Lernkurven für beide neuronalen Netze sind in Abbildung 6.4 dargestellt. Für beide Netze wird zusätzlich eine 5-fache Kreuzvalidierung (*5-fold Cross Validation*) durchgeführt. Hier werden zunächst zufällig gewählte 20% der Lerndaten als Testdaten definiert und die restlichen 80% der Daten zufällig in fünf Teilmengen unterteilt. Das Training der Netze wird dann fünfmal durchgeführt und jeweils vier der Teilmengen werden als Trainings-



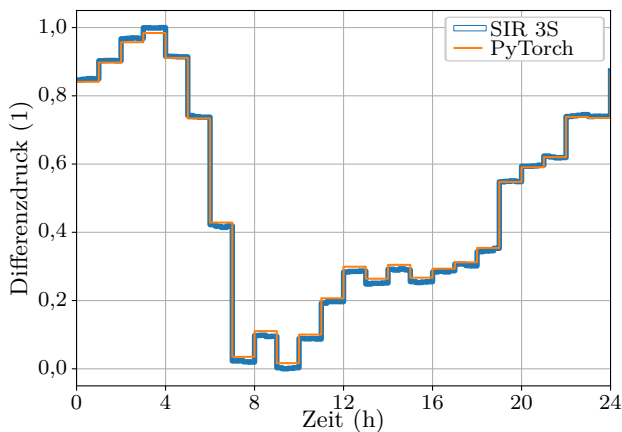
a) Aggregiertes Netzmodell



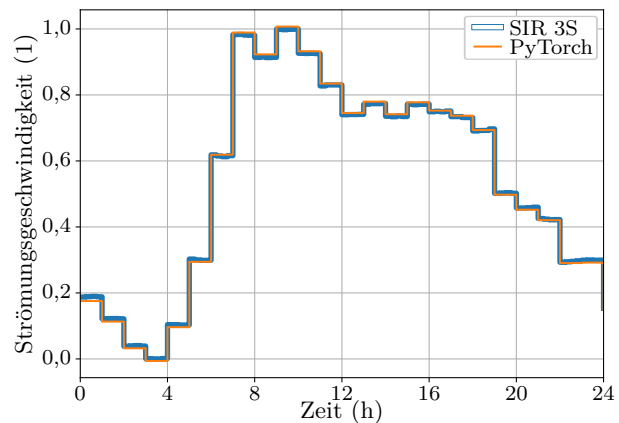
b) Vollständiges Netzmodell

Abbildung 6.4.: Lernkurven für die neuronalen Netze von Fernwärmenetz 1.

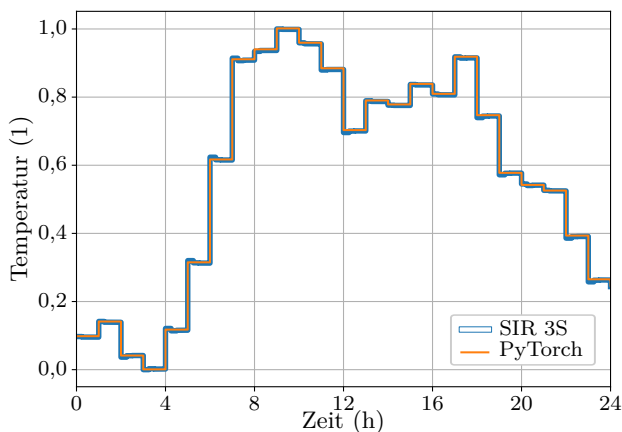
und eine Teilmenge für die Validierungsdaten verwendet. So kann abgeschätzt werden, ob die Modelle besonders gut an bestimmte Bereiche der Lerndaten angepasst bzw. überangepasst sind. In diesem Fall besteht zwischen den Ergebnissen bezüglich der Validierungsdaten der fünf mit unterschiedlichen Teilmengen trainierten Modelle nur ein sehr geringer Unterschied. Eine Überanpassung kann also nicht festgestellt werden.



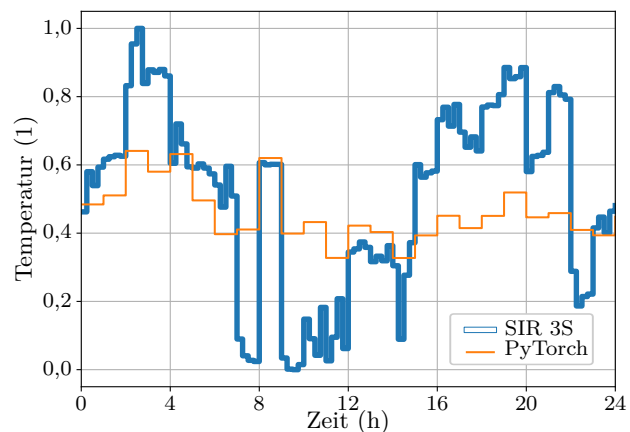
a) Minimaler Differenzdruck an Tag 1



b) Max. Strömungsgeschwindigkeit an Tag 1



c) Minimale Vorlauftemperatur an Tag 1



d) Minimale Vorlauftemperatur an Tag 3

Abbildung 6.5.: Vergleich des neuronalen Netzes (*PyTorch*) von Fernwärmenetz 1 mit *SIR 3S*[®]. Alle Größen sind normiert dargestellt.

Nun werden die Ergebnisse der final trainierten neuronalen Netze bezüglich der Testdaten und Beispieltage vorgestellt und ausgewertet. Die Abbildungen 6.5a und 6.5b enthalten beispielhaft die Berechnungen des minimalen Differenzdrucks und der maximalen Strömungsgeschwindigkeit mit dem neuronalen Netz (*PyTorch*) und dem aggregierten *SIR 3S*[®] Netzmodell für Beispieltag 1 im Januar. Beide Größen werden vom neuronalen Netz genau berechnet. Die Abbildungen 6.5c und 6.5d zeigen die Simulationsergebnisse der minimalen Vorlauftemperatur für Tag 1 und Tag 3. Für Tag 1 ist die Übereinstimmung sehr gut, während das neuronale Netz für Tag 3 eine relativ konstante Temperatur vorhersagt, welche nicht der *SIR 3S*[®] Simulation entspricht. Die Berechnung dieser Temperatur an Tagen mit niedrigem Wärmeverbrauch funktioniert bislang offenbar nicht gut.

Die Tabellen A.5 und A.6 im Anhang geben die maximalen absoluten und relativen Abweichungen für jede Ausgangsgröße des neuronalen Netzes für die vier Beispieltage und das aggregierte Fernwärmenetz an. In den meisten Fällen ist die maximale relative Abweichung kleiner 1%. Relative Abweichungen größer 3% treten für die Pumpenleistung, die minimale Strömungsgeschwindigkeit und die minimale Vorlauftemperatur an Tag 3 im Juli sowie für die minimale Vorlauftemperatur an Tag 4 im Mai auf. Beide Tage weisen niedrige Wärmeverbräuche und somit niedrige Massenströme und Strömungsgeschwindigkeiten auf. In der Software *SIR 3S*[®] macht sich dies durch längere Rechenzeiten bemerkbar, welche unter anderem durch wechselnde Strömungsrichtungen verursacht werden und somit zu langsamerer Konvergenz der Lösungsalgorithmen führen. Hier könnte untersucht werden, ob zusätzliche Trainingsdaten von Simulationen mit niedrigem Wärmeverbrauch die neuronalen Netze verbessern.

Die Tabellen A.7 und A.8 im Anhang zeigen zusätzlich die maximalen absoluten und relativen Abweichungen des neuronalen Netzes für das vollständige *SIR 3S*[®] Netzmodell. Für den Wintertag funktioniert die Annäherung mit einem neuronalen Netz sehr gut. Für Tag 2 im April treten zum Beispiel für den minimalen Differenzdruck teils hohe Abweichungen auf, welche auf einen mittlerweile behobenen Fehler im Interface zur Software *SIR 3S*[®] zurückzuführen sind. Für Tag 3 und 4 tritt dieser Fehler ebenfalls auf und der Vergleich zwischen neuronalen Netzen und *SIR 3S*[®] ist nicht aussagekräftig. Die Trainingsdaten können mit einer aktualisierten Softwareversion neu erzeugt werden, sodass dann keine großen Abweichungen zwischen neuronalen Netzen und *SIR 3S*[®] mehr auftreten sollten.

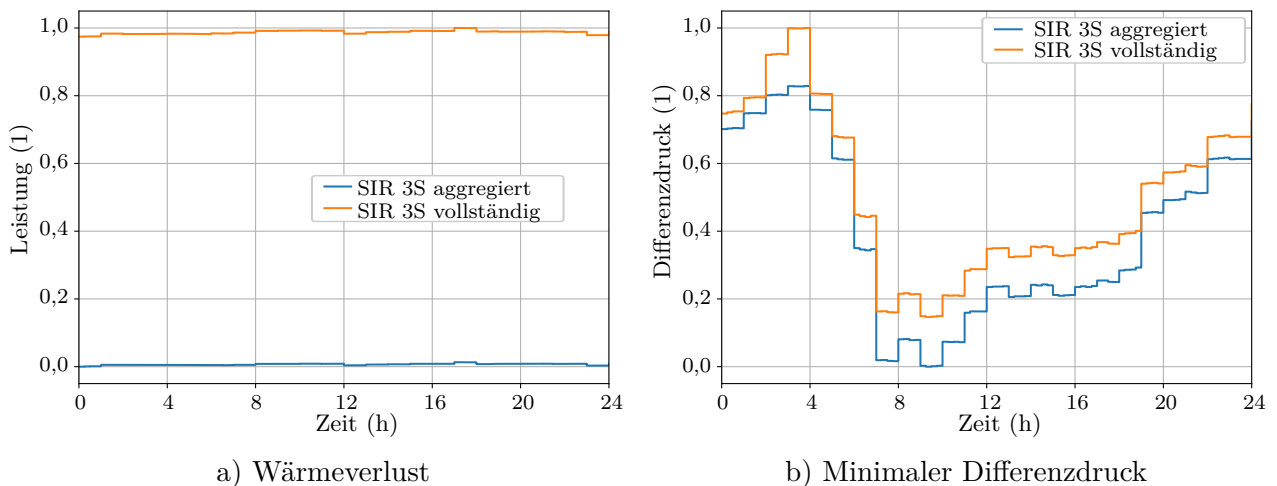


Abbildung 6.6.: Vergleich von normierten Simulationsergebnissen von Beispieltag 1 im Januar mit dem aggregierten und vollständigen *SIR 3S*[®] Netzmodell.

Abbildung 6.6 zeigt des Weiteren einen Vergleich der Wärmeverluste und minimalen Differenzdrücke des aggregierten und vollständigen *SIR 3S*[®] Netzmodells. Hier wird ersichtlich, dass die Aggregation des Netzes Drücke geringfügig absenkt und Wärmeverluste fast halbiert. Vor einer Produktivsetzung des Automatisierungskonzepts dürfte das ausgewählte Modell also nur eingeschränkt aggregiert werden, um Netzzustände korrekt abbilden zu können. Zusätzlich sollte das Modell nicht nur hydraulisch,

sondern auch thermisch kalibriert werden, um solche Abweichungen zu vermeiden.

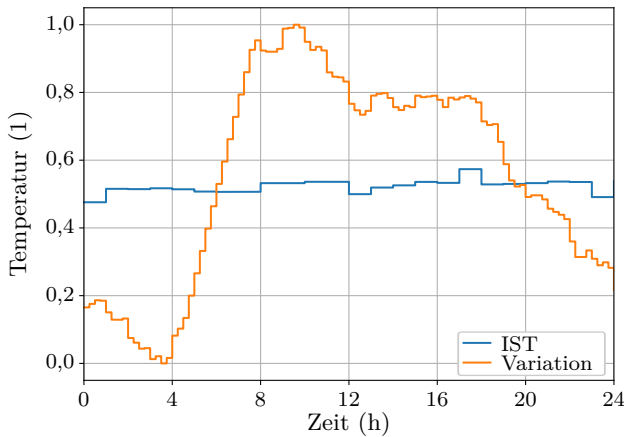
Insgesamt wurde in diesem Kapitel ein Vergleich unterschiedlicher Algorithmen des maschinellen Lernens für die Vereinfachung stationärer Simulationsmodelle von Fernwärmenetz 1 vorgestellt. Neuronale Netze zeigen die besten Ergebnisse und wurden deshalb für detaillierte Untersuchungen ausgewählt. Die Parameter und Architektur der Netze wurden so gewählt, dass das thermo-hydraulische Netzverhalten komplexer *SIR 3S*[®] Modelle möglichst genau bei geringen Rechenzeiten abgebildet wird.

6.1.4. Netz 1: Vereinfachte instationäre Modelle

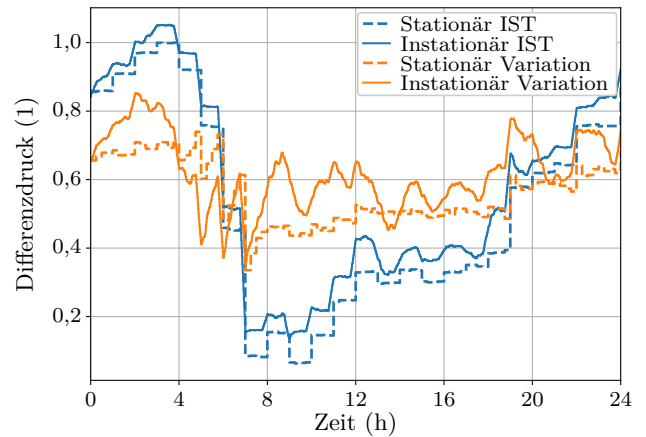
Bisher wurden alle Simulationen quasi-stationär durchgeführt. Die Temperaturlausbreitung wird instantan simuliert. Bei Netz 1 handelt es sich bereits um ein großes Fernwärmenetz mit mehreren Kilometern Leitungslänge vom Wärmeerzeugungsstandort zu den entferntesten Wärmetauschern. Die Validierung der Simulationen in *SIR 3S*[®] durch einen Messdatenvergleich hat gezeigt, dass quasi-stationäre Simulationen das tatsächliche instationäre Netzverhalten an vier Beispieltagen gut abbilden können. Auch instationäre Simulationen in *SIR 3S*[®] stimmen für die simulierten Tage mit den stationären Simulationen dieser Tage überein, wie beispielsweise ein Vergleich der IST-Verläufe in Abbildung 6.7 zeigt. Allerdings wird das Netz in der Realität mit nahezu konstanten Vorlauftemperaturen gespeist. Größere Änderungen der Vorlauftemperatur über einen Tag, wie zum Beispiel durch eine der Wärmeabnahme angepasste Vorlauftemperaturoptimierung, wurden so nicht untersucht und sollen hier zunächst durch eine instationäre Simulation in *SIR 3S*[®] überprüft werden. Bei dieser instationären Simulation handelt es sich um eine thermisch instationäre Betrachtung, welche mit dem gleichen aggregierten Netzmodell durchgeführt wird wie die stationären Simulationen.

Die Vorlauftemperatur von Beispieltag 1 und eine Variation dieser entsprechend der Wärmeabnahme sind in Abbildung 6.7a dargestellt und werden jeweils für eine stationäre Simulation mit 15-Minuten-Zeitschritten und eine instationäre Simulation mit einem Zeitschritt von einer Minute verwendet. Die Ergebnisse für die minimalen Differenzdrücke, die maximalen Strömungsgeschwindigkeiten und die Wärmeeinspeisung sind ebenfalls in Abbildung 6.7 dargestellt. Es kann beobachtet werden, dass die Werte in der instationären Simulation mit der IST-Vorlauftemperatur weniger von der stationären Simulation abweichen als die Werte mit der Vorlauftemperaturvariation. Eine stärkere Änderung der Vorlauftemperatur führt also auch zu größeren Abweichungen stationärer Simulationen von instationären Simulationen bzw. dem tatsächlichen Netzverhalten. In diesem Fall sind stationäre Simulationen nicht ausreichend, um das Netzverhalten korrekt abzubilden.

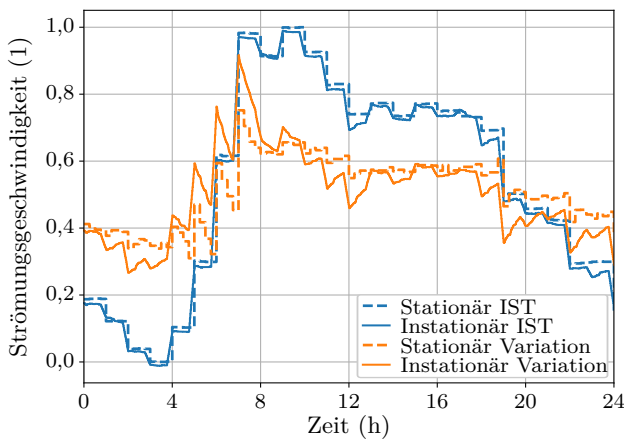
Für Netz 1 soll somit untersucht werden, auch thermisch instationäre Simulationen durch schnellere mathematische Modelle anzunähern. Die Nachbildung der Netzmodelle wurde mit neuronalen Netzen und rekurrenten neuronalen Netzen (RNN) untersucht, hat aber bisher nicht zu ausreichend guten Ergebnissen geführt. Stattdessen wird nun die Systemidentifikation mit der *MATLAB*[®] Systemidentifikationstoolbox in der Version *R2020b* als Alternative für die instationäre Simulation von Netz 1 vorgestellt. Die Theorie der Systemidentifikation wurde in Kapitel 2.4 beschrieben. Gängige Ansätze der Systemidentifikation sind die Identifikation von Übertragungsfunktionen oder Zustandsraummodellen. So lassen sich lineare Modelle identifizieren, für welche direkt die Differenzialgleichungen bekannt sind und somit Ableitungen gebildet werden können. Für gradientenbasierte Optimierungsverfahren wäre dies ein großer Vorteil. Für Netz 1 wird daher zunächst versucht, lineare Modelle für wichtige Parameter wie beispielsweise die Wärmeeinspeisung, Pumpenleistung und maximale Strömungsgeschwindigkeit zu identifizieren. Wie zu erwarten, zeigen die linearen Modelle allerdings keine guten Ergebnisse für die Approximation des nicht-linearen Verhaltens eines Fernwärmenetzes. Anschließend werden nicht-lineare *NonLinear AutoRegressive with eXogenous input* (nlarx) Modelle getestet.



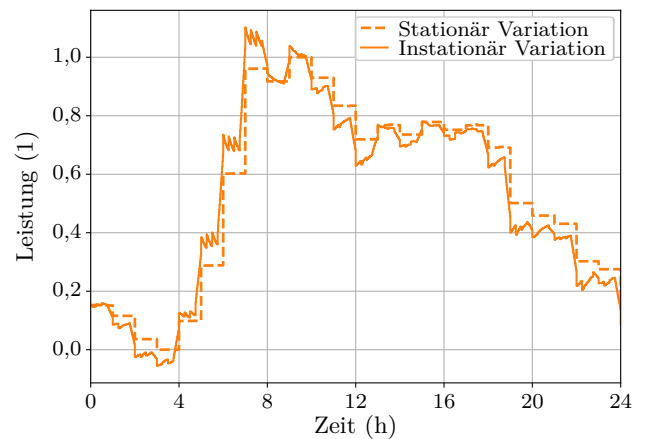
a) IST- und variierte Vorlauftemperatur



b) Minimaler Differenzdruck



c) Maximale Strömungsgeschwindigkeit



d) Wärmeeinspeisung

Abbildung 6.7.: Quasi-stationäre und instationäre *SIR 3S*[®] Simulation von Beispieltag 1 im Januar mit IST- und variierte Vorlauftemperatur. Alle Größen sind mit Versatz der *y*-Achse dargestellt.

Für die Systemidentifikation der Modelle werden die folgenden Simulationsfälle als Basis gewählt, um den Ansatz prinzipiell zu testen:

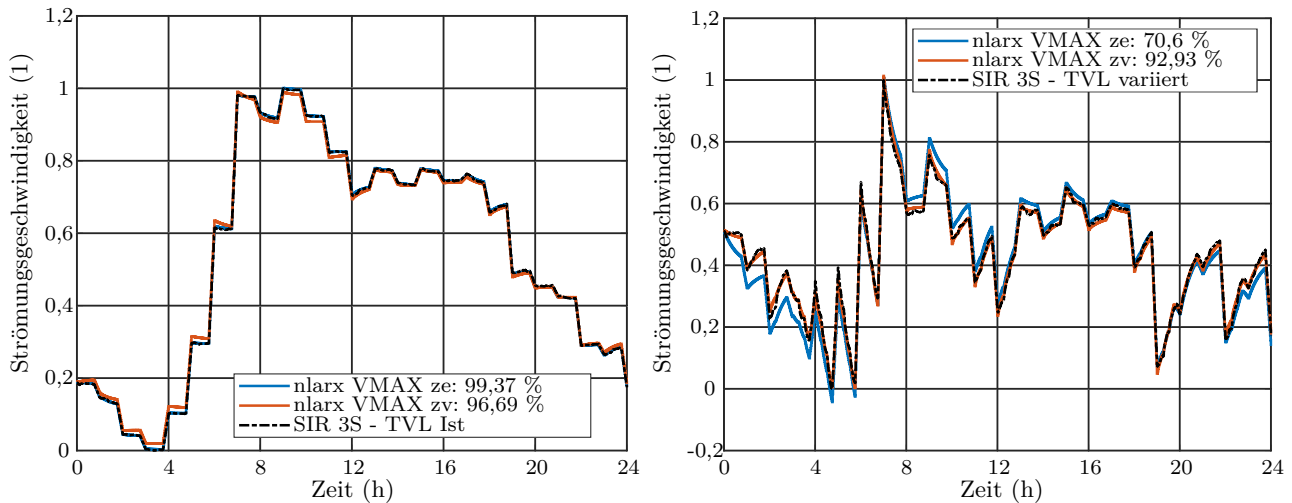
- Wärmebedarf des Wintertages (Beispieltag 1), keine weitere Variation.
- Der Vorlaufdruck wird wie in den Validationsrechnungen mit *SIR 3S*[®] konstant belassen.
- Es werden ein Fall mit konstanter IST-Vorlauftemperatur des Wintertages und ein Fall mit variierte Vorlauftemperatur, dargestellt in Abbildung 6.7a simuliert.

Einer der Temperaturfälle wird jeweils für die Erstellung und ein Fall für die Validierung der Modelle verwendet. Die Simulationen werden mit 1-Minute-Zeitschritten für einen Zeitraum von 24 Stunden durchgeführt. Die Ergebnisse der Systemidentifikation der maximalen Strömungsgeschwindigkeit sind in Abbildung 6.8 dargestellt. Für die Systemidentifikation des Modells *nlarx VMAX ze* wird der Simulationsfall mit der konstanten Vorlauftemperatur verwendet, für das Modell *nlarx VMAX zv* der Fall mit variierte Vorlauftemperatur. Beide Modelle bilden die Strömungsgeschwindigkeit gut ab, wobei das Modell *nlarx VMAX zv* für den Fall mit variierte Vorlauftemperatur deutlich bessere Ergebnisse liefert. In den Abbildungen ist in der Legende jeweils der *Modellfit* in Prozent angegeben. Der *Modellfit* wird mit dem *Normalized Root Mean Squared Error* (NRMSE) folgendermaßen nach Gleichung (6.6) berechnet [123]:

$$NRMSE(y, \hat{y}) = 100 \left(1 - \frac{\|y - \hat{y}\|_2}{\|y - \bar{y}\|_2} \right) \quad (6.6)$$

Das Modell *nlarx VMAX zv* besteht aus zwei Eingangs-Regressoren für den ersten Eingang (Wärmeabnahme), zehn Regressoren für den zweiten Eingang (Vorlauftemperatur) und einem Regressor für den Ausgang. Es werden durch die Systemidentifikationstoolbox also die Parameter der folgenden Funktion optimiert:

$$y_p(t) = F(y(t-1), u_1(t), u_1(t-1), u_2(t), u_2(t-1), \dots, u_2(t-9),) \quad (6.7)$$



a) Simulation mit IST-Vorlauftemperatur

b) Simulation mit variierter Vorlauftemperatur

Abbildung 6.8.: Normierte Ergebnisse für die Systemidentifikation der maximalen Strömungsgeschwindigkeit in Fernwärmenetz 1 mit IST und variierter Vorlauftemperatur.

Für die Wärmeeinspeisung und Pumpenleistung wird auf die gleiche Art und Weise versucht, Modelle zu ermitteln. Zwei Fälle reichen jedoch nicht für eine akzeptable Modellgüte aus. Deshalb werden weitere Temperaturvariationen simuliert, um einen größeren Operationsbereich von Netz 1 abzubilden. Die gewählte Temperaturkurve ist in Abbildung 6.9 zu sehen.

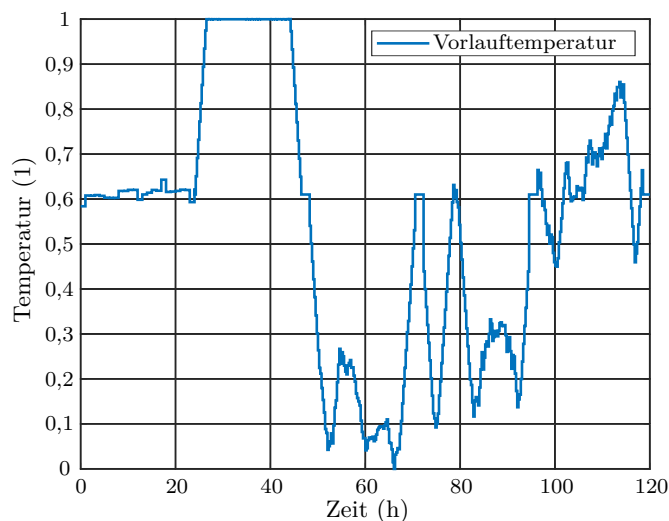


Abbildung 6.9.: Für die Systemidentifikation der Wärmeeinspeisung und Pumpenleistung verwendete normierte Vorlauftemperaturen.

Die Kurve entspricht 5 Tagen mit dem Wärmebedarf von Tag 1 und unterschiedlichen Vorlauftemperaturen. Die Ergebnisse mit den identifizierten nlarx-Modellen für die Wärmeeinspeisung und Pumpenleistung zeigen die Abbildungen 6.10a und 6.10b. Eine weitergehende Validierung der Modelle wird nicht durchgeführt, da der Ansatz der Systemidentifikation nur prinzipiell getestet und auf einige Optimierungsbeispiele angewandt wird. In Kapitel 7.1.3 werden allerdings noch von den Trainingsdaten unabhängige Rechnungen gezeigt, für welche die Modelle ebenfalls gute Ergebnisse liefern. Die nlarx-Modelle bestehen jeweils aus einem Ausgangsregressor und drei und vier Regressoren für die beiden Eingänge der Wärmeeinspeisung sowie zwei und vier Regressoren für die Pumpenleistung.

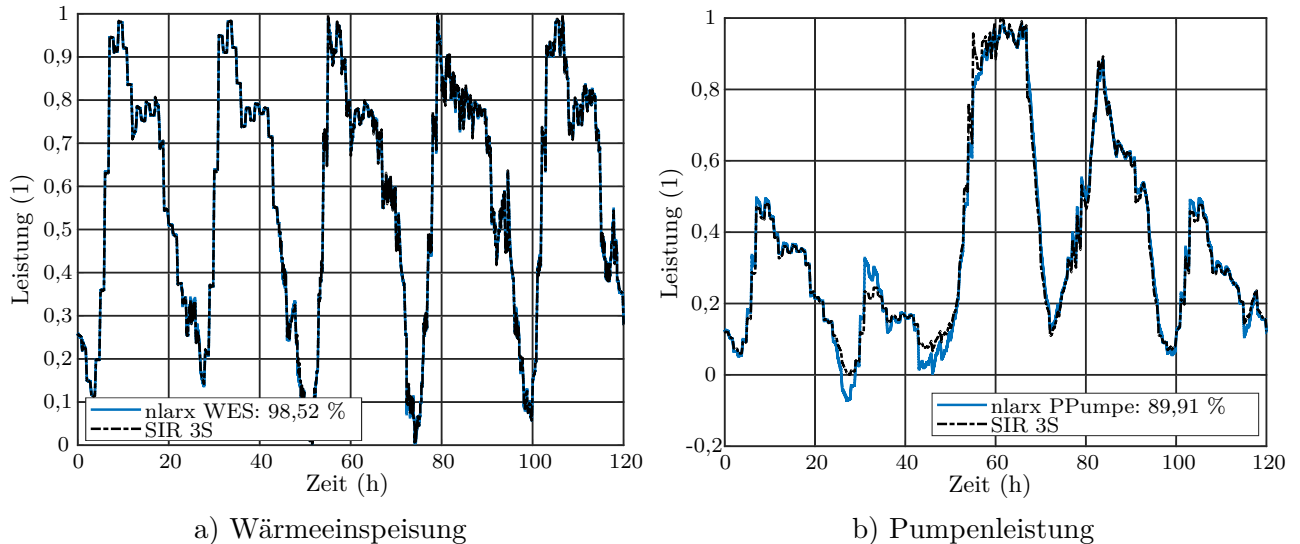


Abbildung 6.10.: Normierte Ergebnisse der Systemidentifikation für die Wärmeeinspeisung und Pumpenleistung.

Des Weiteren ist zu beachten, dass instationäre Simulationen in *SIR 3S*[®] immer in einem stationären Zustand oder mit gegebenen Temperaturdaten in jeder Leitung initialisiert werden. In dieser Arbeit wird eine stationäre Initialisierung zum Zeitpunkt $t = 0$ eingesetzt. Die nlarx-Modelle müssen dementsprechend ebenfalls mit den stationären Startwerten initialisiert werden. Für die initiale stationäre Simulation werden wieder neuronale Netze verwendet, welche in diesem Fall mit der *MATLAB*[®] *Deep Learning Toolbox* trainiert werden. Die *Toolbox* kommt hier zum Einsatz, da die Systemidentifikation bereits in *MATLAB*[®] durchgeführt wurde und so, neben neuronalen Netzen in *PyTorch* eine weitere Möglichkeit für das Training von neuronalen Netzen getestet werden kann. In *MATLAB*[®] werden die gleichen Trainingsdaten und Parameter wie in Kapitel 6.1.3 verwendet.

Abbildung 6.11 zeigt eine quasi-stationäre Simulation der maximalen Strömungsgeschwindigkeit als Beispiel. Auch mit der *MATLAB*[®] *Deep Learning Toolbox* können schnell geeignete Modelle für die Approximation von Fernwärmenetz 1 gefunden werden. Die Erstellung der Netze wird nicht mehr im Detail ausgeführt, da sie bereits für die Netze in *PyTorch* beschrieben wurde.

Die Rechenzeit der Modelle wird zuletzt auf einem Rechner mit *Intel Core i7-4771* (4 x 3,50GHz) und 32 GB RAM durch 100 Simulationen eines Tagesverlaufs mit einem Zeitschritt von einer Minute in Schleife bestimmt. Inklusiv der stationären Initialisierung dauert eine Berechnung der maximalen Strömungsgeschwindigkeit im Schnitt 150 ms, eine Simulationen der Wärmeeinspeisung 122 ms und der Pumpenleistung 118 ms. Dem gegenüber stehen Rechenzeiten in *SIR 3S*[®] von etwa 30 s für Wintertage und mehreren Minuten für Tage im Sommer.

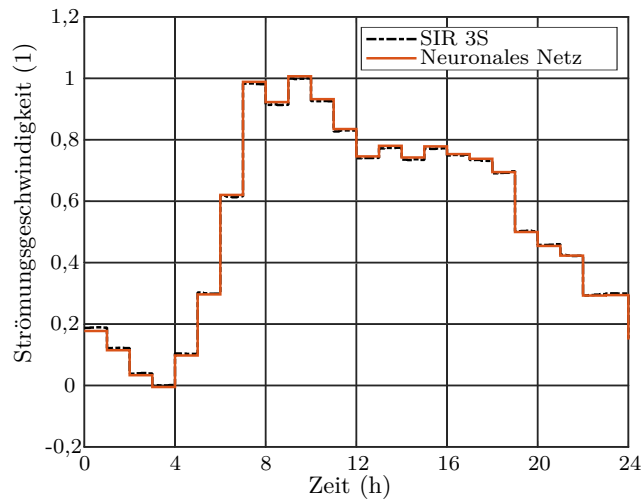


Abbildung 6.11.: Normierte Ergebnisse der maximalen Strömungsgeschwindigkeit an Tag 1 im Januar mit IST-Vorlauftemperatur und der *MATLAB[®] Deep Learning Toolbox*.

Insgesamt konnte in diesem Kapitel gezeigt werden, dass stationäre Simulationen mit zunehmend starker Variation der Vorlauftemperaturen auch zunehmend von instationären Simulationen bzw. dem realen Fernwärmenetzverhalten abweichen. Für Optimierungen der Vorlauftemperatur ist somit unter Umständen ein instationäres Simulationsmodell erforderlich. Mit dem Ansatz der Systemidentifikation wurden geeignete instationäre Modelle mit Simulationszeiten von unter 200 ms für einige beispielhaft gewählte hydraulische Größen von Fernwärmenetzen gefunden. Mit der *MATLAB[®] Deep Learning Toolbox* ist es zudem ebenso wie mit *PyTorch* möglich, akkurate neuronale Netze für stationäre Simulationen zu trainieren. Eine detaillierte Modellerstellung wurde bislang nicht durchgeführt, weshalb keine allgemeinen Abweichungen der Systemidentifikationsmodelle zu *SIR 3S[®]* angegeben werden. Der Ansatz kann für genauere instationäre Simulationen und Optimierungen weiterverfolgt werden.

6.2. Fernwärmenetz 2

Abbildung 6.12 zeigt ein Schema von Fernwärmenetz 2, welches von einem HKW und einem HW gespeist wird. Das HKW, an dessen Standort auch Wärme aus einer Geothermieanlage eingespeist wird, ist hauptsächlich für die Fernwärmeversorgung verantwortlich, während das HW bei hohen Abnahmelasten hinzugeschaltet wird. In dem Netz ist ebenfalls eine DEA etwa auf der Hälfte der Distanz zwischen HKW und HW installiert, welche den Rücklaufdruck auf dem Weg zurück zum HKW erhöhen kann. Verglichen mit Fernwärmenetz 1 ist Netz 2 deutlich größer und bietet mit zwei vorhandenen Wärmeerzeugungsanlagen zudem die Möglichkeit, eine verteilte Wärmeversorgung zu analysieren. Die vorgestellten Modellierungsansätze können schrittweise auf eine komplexere Aufgabenstellung angewandt werden. Im Folgenden wird zunächst wieder ein thermo-hydraulisches Modell in *SIR 3S*[®] vorgestellt, für welches anschließend Modellvereinfachungen untersucht werden.

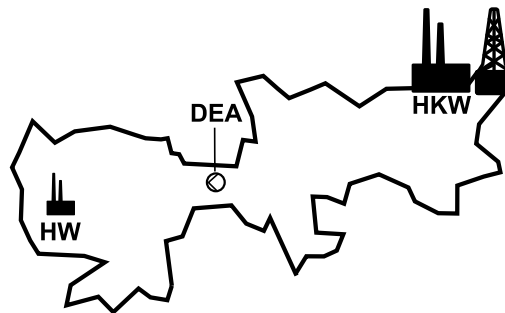


Abbildung 6.12.: Vereinfachte, nicht-maßstäbliche Darstellung des realen Fernwärmenetzes 2 mit Druckerhöhungs- und Wärmeerzeugungsanlagen.

6.2.1. Netz 2: *SIR 3S*[®]

Netz 2 wird von der 3S Consult GmbH vollständig mit etwa 18000 Knoten und 17000 Leitungen sowie aggregiert mit etwa 11000 Knoten und 10000 Leitungen modelliert. Bei der Aggregation des Netzes wird aufgrund der Erfahrung mit Netz 1 explizit darauf geachtet, relevante Berechnungsgrößen des Netzes wie Wärmeverluste und Drücke nicht zu verändern. Der Grad der Aggregation fällt entsprechend geringer aus. Während Netz 1 auf etwa 20 % der Knoten aggregiert werden konnte, wird Netz 2 nur auf 60 % aggregiert. In diesem Fall werden auch die Leitungsgesamtlänge und Anzahl der Wärmeübergabestationen nicht reduziert. Die zwei Wärmeerzeugungsanlagen sind im Netzmodell vereinfacht als einzelne Anlagen mit Umwälzpumpen modelliert. Für das HKW steht auch ein separates erweitertes Modell inklusive der Geothermieanlagen zur Verfügung. Dieses wird im Folgenden nicht im Detail behandelt, da es für spätere Optimierungen nicht notwendig ist. Für die *SIR 3S*[®] Netzmodelle ergeben sich folgende Eingangsvariablen:

- Gesamtmassenstrom
- Massenstrom und Vorlauftemperatur des HW
- 2 Mindestdrücke im Vorlauf und 8 Maximaldrücke im Rücklauf, welche anschließend von dem *SIR 3S*[®] Modell geregelt werden.
- Druckhaltung, Vor- und Rücklauftemperatur des HKW. Der Massenstrom ergibt sich aus der Differenzdruckregelung, welche im Modell implementiert ist.
- Druckerhöhung durch die DEA im Rücklauf

Auch hier ist die Rücklauftemperatur für eine Prognoseberechnung nicht bekannt und muss entweder separat vorhergesagt oder als konstant angenommen werden. Für eine Kalibrierung und Validierung des Modells werden Nachberechnungen der folgenden drei Beispieltage in 15-Minuten-Schritten genutzt:

- Tag 1: Hochlastfall im Februar 2021
- Tag 2: Fall in der Übergangszeit im Mai 2020
- Tag 3: Niedriglastfall im Juli 2021

Die Betriebsweise des Netzes wird dabei so modelliert, dass Druck- und Temperaturegebnisse möglichst den Messwerten entsprechen. Für eine gute Übereinstimmung der Temperaturen werden die Wärmedurchgangskoeffizienten der Rohre sowie die Verteilung der Rücklauftemperaturen angepasst. Die Differenzdruckregelung wird nicht exakt wie in der Realität nachgebildet, sondern durch die vorab genannten 2 Druckmesspunkte im Vorlauf und 8 Punkte im Rücklauf modelliert, welche an jedem Beispieltag konstant eingestellt sind. Auch die Druckhaltung bleibt an den Beispieltagen konstant. Wärmeerzeugungsanlagen inklusive deren Rücklauftemperaturen werden entsprechend der Messdaten an den drei Tagen eingestellt. Daraus ergeben sich die mittleren 24-Stunden-Druck- und Temperaturabweichungen zu den verfügbaren Messwerten in Tabelle 6.10. Eine quasi-stationäre 24-Stunden-Simulation mit 15-Minuten-Zeitschritten dauert mit dem Modell wenige Minuten. Tage mit hohen Wärmeleistungen können schneller simuliert werden als Tage mit niedriger Einspeisung.

Tabelle 6.10.: Mittlere 24h-Abweichungen der Nachberechnung von Fernwärmenetz 2 in *SIR 3S*[®] an den drei Beispieltagen.

	Tag 1	Tag 2	Tag 3
Druck (bar)	0,17	0,24	0,36
Standardabweichung Druck (bar)	0,14	0,22	0,22
Druck (%)	4,5	7,2	8,3
Temperatur (°C)	0,99	0,87	3,09
Standardabweichung Temperatur (°C)	1,37	1,63	3,3
Temperatur (%)	1,3	1,2	5,3

6.2.2. Netz 2: Maschinelles Lernen

Nun soll die Anwendbarkeit von Ansätzen des maschinellen Lernens für Fernwärmenetz 2 untersucht werden. Das Ziel ist wieder die Rechenzeiten zu reduzieren und das Simulationsmodell in *SIR 3S*[®] möglichst genau zu approximieren. Für Fernwärmenetz 1 zeigten neuronale Netze die besten Ergebnisse. Diese Erkenntnis soll hier mit einem etwas komplexeren Netz verifiziert werden.

Die Analysen werden mit dem aggregierten Netzmodell und quasi-stationären Simulationen untersucht. Insgesamt sind 2933 145 stationäre Simulationsergebnisse aus *SIR 3S*[®] mit variierten Eingangsgrößen verfügbar. Die Eingangsgrößen in *SIR 3S*[®] werden über den gesamten an den drei Beispieltagen gemessenen Betriebsbereich variiert. Ebenfalls werden jeweils etwas niedrigere und höhere Eingangswerte verwendet, um potenziell niedrigere oder höhere Wärmeverbräuche abzubilden. Die Druckhaltung des HKW sowie die Druckregelziele an den Schlechtpunkten im Vor- und Rücklauf werden nicht variiert und können nur mit den Standardwerten entsprechend der Modellvalidierung in *SIR 3S*[®] abgebildet werden. Mit diesen Standardwerten wurde für die drei Beispieltage eine gute Übereinstimmung zwischen *SIR 3S*[®] Rechenwerten und Messwerten erreicht. Des Weiteren wird die Druckerhöhung der DEA nur mit 0 bar und 1 bar simuliert. An den Beispieltagen wurde die Anlage nicht verwendet. Die Rücklauftemperaturen werden wie an den Beispieltagen gemessen simuliert. Durch die Variation der Eingangsdaten kann insgesamt auch der Betriebsbereich von Netz 2 fast vollständig abgedeckt werden. Vorab anzumerken ist, dass das Verhalten der DEA mit den zwei simulierten Druckerhöhungen

nicht ausreichend nachgebildet werden kann. Für ein gutes Lernergebnis sind weitere Simulationen notwendig.

Alle ungültigen Simulationen werden vorab von der Software *SIR 3S*[®] oder in einer nachgehenden Analyse identifiziert. Dabei werden alle Simulationsergebnisse mit Drücken oder Differenzdrücken kleiner 0 bar oder Pumpendrehzahlen kleiner 0 min^{-1} gelöscht. Zusätzlich werden Simulationen mit Pumpendrehzahlen im Heizwerk größer der Maximaldrehzahl und minimale Vorlauftemperaturen kleiner 20°C entfernt. Beide Fälle treten selten auf und werden als fehlerhafte Simulationen betrachtet. Für die minimale Vorlauftemperatur tritt wie bei Netz 1 teilweise ein Berechnungsfehler im Interface zu *SIR 3S*[®] auf, welcher im Nachgang behoben wurde. Die *SIR 3S*[®] Modelle liefern nun also korrekte Ergebnisse, während für das Lernen noch die fehlerhaften, größtenteils bereinigten Simulationsdaten der alten Softwareversion verwendet wurden.

Für die Anwendung in Modellen des maschinellen Lernens wird statt des Gesamtmassenstroms die gesamte Wärmeleistung als Eingang verwendet. Somit kann das Modell basierend auf Wärmeeinspeisungsprognosen für Prognosesimulationen verwendet werden. Der Gesamtmassenstrom ist eine Ausgangsgröße der Algorithmen. Zudem werden jeweils Modelle mit und ohne Rücklauftemperatur als Eingang trainiert. So werden insgesamt sechs bzw. fünf Eingangsgrößen verwendet: die gesamte Wärmeleistung, die Erzeugungsleistung des HW, die Vorlauftemperaturen am HKW und HW, die Druckerhöhung durch die DEA sowie die optionale Rücklauftemperatur am HKW.

Modelle werden anschließend mit den folgenden 24 Ausgangsgrößen trainiert, welche später in der Optimierung als Zielgrößen oder Restriktionen eingesetzt werden:

- Gesamteinspeiseleistung, Gesamtmassenstrom, Wärmeverbrauch und Wärmeverlust
- Massenströme, Strömungsgeschwindigkeiten, Pumpenleistungen und -Drehzahlen von HKW, HW und der DEA
- Leistung des HKW
- Minimaler Differenzdruck (netzweit) und minimale sowie maximale Drücke im Vor- und Rücklauf (netzweit)
- Minimale Temperatur im Vorlauf (netzweit)

Vergleich verschiedener Ansätze des maschinellen Lernens

Mit den Daten werden nun erneut die in Kapitel 6.1.3 untersuchten Algorithmen in Python 3.8.12 getestet. Dabei wird eine zufällige Unterteilung der Daten in 80 % Trainingsdaten (2 346 516 Datensätze) und 20 % Testdaten (586 629 Datensätze) gewählt. Für neuronale Netze werden zusätzlich 20 % der Daten für die Validierung von Parametern während des Trainings verwendet und somit nur 60 % (1 759 887 Datensätze) für das Training. Wie für Fernwärmenetz 1 werden alle Eingangs- und Ausgangsdaten zwischen 0 und 1 normiert. Die Auswertung der Modelle erfolgt anschließend aufgrund der Bewertungskriterien *Mean Absolute Percentage Error* (MAPE), R^2 und der Rechenzeit (t_s) für stationäre 24-Stunden-Simulationen im 15-Minuten-Takt aus jeweils fünf Rechnungen, durchgeführt auf einem Rechner mit *Intel Core i7-4771* (4 x 3,50GHz) und 32 GB RAM. Die Ergebnisse sind in Tabelle 6.11 (ohne Rücklauftemperatur als Eingang) und Tabelle 6.12 (mit Rücklauftemperatur) dargestellt.

Tabelle 6.11.: Auswertung verschiedener Machine Learning Modelle mit Testdaten von Netz 2, ohne Rücklauftemperatur als Eingang.

Modell	MAPE (%)	R^2	ts (ms)
Lineare Regression	22,181	0,80479	0,45
Polynomregression ($n = 2$)	6,569	0,91987	0,7
Polynomregression ($n = 5$)	2,635	0,96758	1,2

Tabelle 6.11 fortgesetzt.

Modell	MAPE (%)	R ²	ts (ms)
Polynomregression ($n = 7$)	2,351	0,97464	1,6
<i>Random Forests</i>	0,756	0,99197	28
XGBoost (Tiefe = 6)	3,618	0,97557	120
XGBoost (Tiefe = 10)	1,730	0,98799	140
Neuronales Netz ($s = 4, n = 192/96/48/24$)	1,127	0,98151	3

Tabelle 6.12.: Auswertung verschiedener Machine Learning Modelle mit Testdaten von Netz 2, mit Rücklaufftemperatur als Eingang.

Modell	MAPE (%)	R ²	ts (ms)
Lineare Regression	21,410	0,80468	0,45
Polynomregression ($n = 2$)	6,252	0,92046	0,7
Polynomregression ($n = 5$)	2,107	0,96853	1,2
<i>Random Forests</i>	1,184	0,99184	28
XGBoost (Tiefe = 6)	3,155	0,97537	120
XGBoost (Tiefe = 10)	1,878	0,98769	140
Neuronales Netz ($s = 4, n = 192/96/48/24$)	0,549	0,98156	3

Die Ergebnisse entsprechen denen für Netz 1. Während Baumalgorithmen wie *Random Forest* oder XGBoost sehr gute Ergebnisse erzielen, benötigen diese auch die längsten Simulationszeiten. Für die Polynomregression können mit der Rücklaufftemperatur als Eingang maximal Polynome vom Grad 5 trainiert werden, ohne Rücklaufftemperatur vom Grad 7. Für höhere Polynomgrade wird deutlich mehr Arbeitsspeicher benötigt. Die Verwendung der Rücklaufftemperatur als zusätzlichen Eingang zeigt keine signifikante Verbesserung für alle Algorithmen außer den neuronalen Netzen. Hier können die Modellfehler etwa halbiert werden. Insgesamt zeigen neuronale Netze wieder sehr gute Genauigkeiten bei relativ niedrigen Simulationszeiten.

Untersuchung der Anzahl der Schichten und Neuronen

Nun soll ein neuronales Netz ausgelegt werden, welches möglichst geringe Abweichungen zu *SIR 3S*[®] Simulationen aufweist. Dafür wird wie für Netz 1 die Bibliothek *PyTorch* für das Training und das ONNX-Format sowie dessen *Runtime* für die Ausführung der Netze verwendet. Die Anzahl der Ausgänge wird zudem auf 26 erweitert, um eine Berechnung der Vor- und Rücklaufdrücke am HKW in die Modelle aufzunehmen. Für die Auswahl der Anzahl der Schichten und Neuronen werden nur neuronale Netze mit der Rücklaufftemperatur als Eingang trainiert. Die Parameter werden entsprechend den Ergebnissen für Fernwärmenetz 1 folgendermaßen gewählt:

- Optimierer: *Adam* mit einer Lernrate von 10^{-3}
- *Loss*-Funktion: MAE
- Aktivierungsfunktion: Swish
- Initialisierung der Gewichte mit *He*-, bzw. *Kaiming-Uniform*
- 60%/20%/20% Unterteilung und Normierung der Ein- und Ausgangsdaten zwischen 0 und 1.

Tabelle A.9 im Anhang zeigt die Trainingsergebnisse von Netzen mit einer bis vier versteckten Schichten und unterschiedlicher Anzahl an Neuronen nach 100 Epochen. Insgesamt ergibt sich nur ein geringer Unterschied zwischen den Modellen. Am deutlichsten weichen die Ergebnisse für die Ausgangsgrößen Gesamtmassenstrom und minimaler Rücklaufdruck ab. Anhand dieser Größen werden im Anhang Bewertungen vorgenommen und die besten Modelle mit ein bis vier Schichten für weitere Untersuchungen ausgewählt.

Untersuchung der Lernrate

Der Einfluss der Lernrate soll noch genauer analysiert werden. Die gewählten Netzkonfigurationen werden einmal mit konstanter Lernrate und zusätzlich mit einer Halbierung der Lernrate alle 200 Epochen (mit *Decay*) trainiert. Die Ergebnisse sind in der folgenden Tabelle 6.13 dargestellt. Aufgelistet werden der durchschnittliche MAPE und MAE für die Druckausgänge der Netze. Der „MAPE gesamt“ ist dabei der durchschnittliche absolute prozentuale Fehler über 11 der 26 Ausgangsgrößen der neuronalen Netze. Einige der Werte werden für die spätere Optimierung nicht mehr benötigt oder schwanken häufig um 0, wie zum Beispiel die Leistung, der Massenstrom und die Strömungsgeschwindigkeit am Heizwerk, weshalb selbst bei geringen Abweichungen hohe prozentuale Fehler auftreten können. Diese Werte werden folglich in der Berechnung des „MAPE gesamt“ nicht berücksichtigt. Eine Beschreibung der Ausgangsgrößen und welche von diesen für die Berechnung des durchschnittlichen absoluten prozentualen Fehlers verwendet werden, befindet sich in Tabelle A.11 im Anhang. Eine vollständige Auswertung aller Ausgangsgrößen folgt zudem später mit dem final ausgewählten neuronalen Netz. Die in der Tabelle mit t_s bezeichnete durchschnittliche Rechenzeit der Modelle wird aus 100 in Schleife durchgeführten 24-Stunden-Simulationen im 15-Minuten-Takt berechnet.

Tabelle 6.13.: Bewertung neuronaler Netze mit Testdaten für Fernwärmenetz 2 mit Rücklauftemperatur als Eingang nach 1000 Epochen Training.

Neuronen pro Schicht	<i>MAPE</i> Drücke (%)	<i>MAE</i> Drücke (bar)	<i>MAPE</i> gesamt (%)	R^2	t_s (ms)
130	1,891	0,026	1,258	0,96631	0,098
208/156	0,378	0,008	0,286	0,99521	0,174
104/104/52	0,608	0,006	0,372	0,99560	0,137
208/104/52/26	0,294	0,005	0,203	0,99574	0,186
208/156/104/26	0,218	0,005	0,156	0,99586	0,207
208/260/208/156	0,269	0,005	0,198	0,99589	0,417
130 (mit Decay)	1,890	0,029	1,145	0,96418	0,096
208/156 (mit Decay)	0,786	0,007	0,455	0,99498	0,179
104/104/52 (mit Decay)	0,267	0,005	0,163	0,99557	0,136
208/104/52/26 (mit Decay)	0,224	0,005	0,137	0,99572	0,185
208/156/104/26 (mit Decay)	0,175	0,004	0,108	0,99595	0,205
208/260/208/156 (mit Decay)	0,180	0,004	0,102	0,99591	0,422

Alle neuronalen Netze mit einem „MAPE gesamt“ größer 1 % werden bereits an dieser Stelle aussortiert. Die Ergebnisse der übrigen Modelle sind ähnlich und zeigen niedrige Rechenzeiten unter einer halben Millisekunde, weshalb sie im nächsten Schritt wieder auf zwei der Beispieltage angewandt werden. Anzumerken ist, dass, wie eingangs im Abschnitt zur Datenerzeugung erwähnt, Rücklauftemperaturen der Beispieltage in den Trainings- und Testdaten verwendet wurden. Die Ergebnisse für die Beispieltage sind für einen Großteil der Modelle sehr gut. Um sicherzustellen, dass die Modelle nicht aufgrund der im Training verwendeten Rücklauftemperaturen gute Ergebnisse liefern, werden für jeden Beispieltag weitere unabhängige Testsimulationen erstellt. Für jeden Tag werden zehn 24-Stunden-Simulationen mit 15-Minuten-Zeitschritten durchgeführt, in welchen alle Eingangsvariablen in jedem Zeitschritt um die tatsächlich an dem Tag gefahrenen Werte variiert werden. In *SIR 3S*[®] werden Massenströme zufällig um bis zu $\pm 5\%$, Vorlauftemperaturen um bis zu $\pm 3^\circ\text{C}$ und Rücklauftemperaturen um bis zu $\pm 1^\circ\text{C}$ verändert. So ergibt sich auch eine Überschneidung der Eingangsdaten über die drei Beispieltage hinweg. Anschließend werden die zehn Tage mit variierten Eingangsgrößen mit den neuronalen Netzen simuliert und die Ergebnisse mit denen aus *SIR 3S*[®] verglichen. Die Tabellen 6.14 und 6.15 zeigen die Auswertung für den Winter- und den Sommertag. Die Bewertungskriterien sind die gleichen wie zuvor. Zusätzlich ist der maximale relative Fehler von den 11 relevanten der 26 Ausgangsgrößen dargestellt.

Tabelle 6.14.: Abweichung verschiedener neuronaler Netze für Fernwärmenetz 2 zu $SIR\ 3S^{\circledast}$ für zehn Simulationstage basierend auf Tag 1 im Februar.

Neuronen pro Schicht	<i>MAPE</i> Drücke (%)	<i>MAE</i> Drücke (bar)	<i>MAPE</i> gesamt (%)	Max. rel. Error (%)
208/156	0,196	0,007	0,169	1,703
104/104/52	0,138	0,005	0,139	1,120
208/104/52/26	0,092	0,004	0,087	0,971
208/156/104/26	0,082	0,003	0,074	0,697
208/260/208/156	0,093	0,003	0,093	0,958
208/156 (mit Decay)	0,162	0,005	0,129	1,866
104/104/52 (mit Decay)	0,118	0,004	0,082	1,481
208/104/52/26 (mit Decay)	0,088	0,003	0,061	0,959
208/156/104/26 (mit Decay)	0,076	0,003	0,054	0,709
208/260/208/156 (mit Decay)	0,072	0,003	0,049	0,739

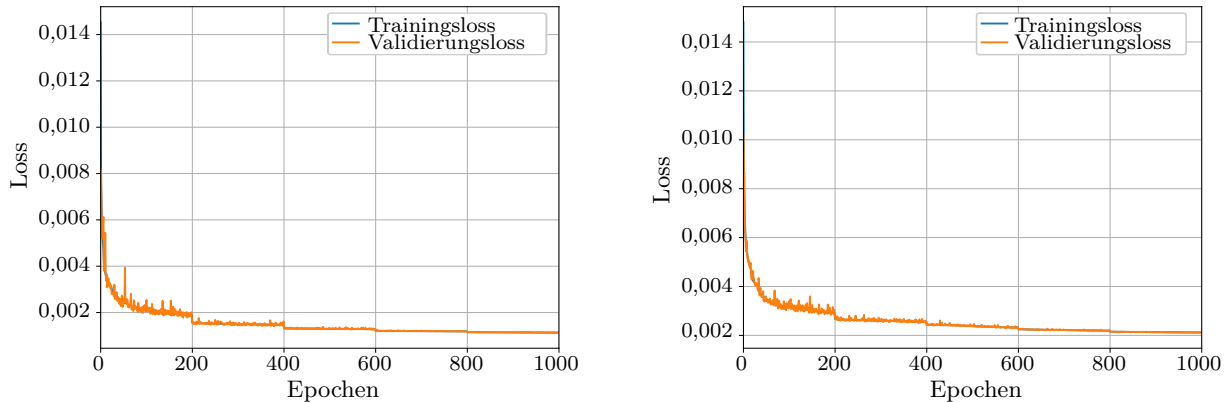
Tabelle 6.15.: Abweichung verschiedener neuronaler Netze für Fernwärmenetz 2 zu $SIR\ 3S^{\circledast}$ für zehn Simulationstage basierend auf Tag 3 im Juli.

Neuronen pro Schicht	<i>MAPE</i> Drücke (%)	<i>MAE</i> Drücke (bar)	<i>MAPE</i> gesamt (%)	Max. rel. Error (%)
208/156	0,169	0,005	0,523	5,339
104/104/52	0,144	0,004	0,385	4,870
208/104/52/26	0,153	0,004	0,196	5,198
208/156/104/26	0,135	0,003	0,355	4,983
208/260/208/156	0,143	0,003	0,262	4,903
208/156 (mit Decay)	0,174	0,005	0,228	4,972
104/104/52 (mit Decay)	0,129	0,003	0,298	5,027
208/104/52/26 (mit Decay)	0,126	0,003	0,355	5,215
208/156/104/26 (mit Decay)	0,131	0,003	0,335	5,200
208/260/208/156 (mit Decay)	0,128	0,003	0,141	5,173

Für den Wintertag in Tabelle 6.14 liefern alle neuronalen Netze gute Ergebnisse. Die geringsten Abweichungen zeigt das größte Modell mit 208, 260, 208 und 156 Neuronen, welches mit abnehmender Lernrate trainiert wurde. Auch für den Sommertag in Tabelle 6.15 liefert das Netz im Vergleich zu den anderen Modellen gute Ergebnisse und wird deshalb ausgewählt. Anzumerken ist noch, dass es mit den in diesem Kapitel ausgewerteten Modellen nicht zu einer deutlich sichtbaren Überanpassung an die Trainingsdatensätze kommt. Es ist also möglich, dass auch größere neuronale Netze noch gute Ergebnisse liefern. Nach der Auswertung im Anhang in Tabelle A.9 ist allerdings erkennbar, dass auch deutlich größere Netze verglichen mit den Testdaten nicht mehr zu einer Verbesserung führen.

Neuronale Netze für Fernwärmenetz 2

Zuletzt werden die Ergebnisse mit dem trainierten neuronalen Netz ausgewertet. Zusätzlich wird ein neuronales Netz ohne die Rücklauftemperatur als Eingang trainiert, um den Einfluss dieser Eingangsvariable zu bestätigen und auch für diesen Fall Ergebnisse auswerten zu können. Der MAPE für die Druckausgänge beträgt für dieses Modell 0,798 %, der MAE 0,007 bar. Über die 11 relevanten Ausgänge beträgt der MAPE 0,514 %. Die Ergebnisse mit dem Modell sind also etwas schlechter verglichen zu dem neuronalen Netz mit Rücklauftemperatureingang. Für die neuronalen Netze für Fernwärmenetze 2 wurde wie für Fernwärmenetz 1 eine 5-fache Kreuzvalidierung durchgeführt, um sicherzugehen, dass keine Überanpassung an bestimmte Teilmengen der Lerndaten vorkommt. Abbildung 6.13 zeigt die Lernkurven der beiden final, einzeln trainierten neuronalen Netze. In diesen sind aufgrund der



a) NN mit Rücklaufftemperatureingang

b) NN ohne Rücklaufftemperatureingang

Abbildung 6.13.: Lernkurven für die neuronalen Netze (NN) von Fernwärmenetz 2.

alle 200 Epochen reduzierten Lernrate Sprünge zu sehen. Zusammenfassend zeigt die folgende Tabelle 6.16 die durchschnittlichen Abweichungen für Drücke und die entsprechend Tabelle A.11 relevanten Ausgangsgrößen für die Testdaten und drei Beispieltage.

Tabelle 6.16.: Durchschnittliche Abweichung der neuronalen Netze zu $SIR\ 3S^{\circledast}$ für Fernwärmenetz 2

	MAPE Drücke (%)	MAE Drücke (bar)	MAPE gesamt (%)
Neuronales Netz mit Rücklaufftemperatur			
Testdaten	0,180	0,004	0,102
Zehn Simulationstage basierend auf Tag 1	0,072	0,003	0,049
Zehn Simulationstage basierend auf Tag 2	0,073	0,003	0,096
Zehn Simulationstage basierend auf Tag 3	0,128	0,003	0,141
Neuronales Netz ohne Rücklaufftemperatur			
Testdaten	0,798	0,007	0,514
Zehn Simulationstage basierend auf Tag 1	0,407	0,018	0,611
Zehn Simulationstage basierend auf Tag 2	0,867	0,035	1,530
Zehn Simulationstage basierend auf Tag 3	0,164	0,004	0,616

Die Tabellen A.12 bis A.14 im Anhang enthalten zudem die maximalen absoluten und relativen Abweichungen zu $SIR\ 3S^{\circledast}$ für alle Ausgangsgrößen der neuronalen Netze mit und ohne Rücklaufftemperatur und für die jeweils zehn Simulationen basierend auf drei Beispieltagen. Die Tabellen 6.17 und 6.18 zeigen die Ergebnisse für alle durchschnittlichen relativen Abweichungen größer 3% (rot markiert) für den Winter- und Sommertag. In Tabelle 6.17 ist ersichtlich, dass das neuronale Netz mit Vorlaufftemperatur als Eingangsvariable an einem Wintertag nur bei 4 der 26 Ausgangsgrößen maximale relative Abweichungen größer 3% aufweist. Die durchschnittlichen relativen Abweichungen sind für alle Ausgangsvariablen außer der Drehzahl der DEA kleiner 1%. Die Pumpenleistung und -drehzahl der DEA weichen stark ab, da diese die gesamte Zeit ausgeschaltet ist, das neuronale Netz aber kleine absolute Abweichungen voraussagt, welche hohe relative Abweichungen verursachen. Diese Abweichungen sind daher grau markiert. Die minimale Vorlaufftemperatur zeigt hohe relative Abweichungen aufgrund des inzwischen behobenen Fehlers im Interface zu $SIR\ 3S^{\circledast}$, welcher falsche Ausgaben der Temperatur verursachte. Die durchschnittliche Abweichung liegt bei nur 0,585%. Für die Vorlaufpumpen des HKW werden zu einigen Zeitpunkten Drehzahlabweichungen bis zu 17,45% berechnet. Diese werden durch einzelne Ausreißer des neuronalen Netzes verursacht. Der Drehzahlbereich ist allerdings weit von den Minimal- oder Maximaldrehzahlen entfernt, weshalb in einer Optimierung keine Probleme auftreten sollten. Dies muss aber noch in Tests am realen System nachgewiesen werden.

Das neuronale Netz ohne Rücklauftemperatur zeigt deutlich höhere Abweichungen und 16 Ausgänge zeigen maximale relative Abweichungen größer 3%. Durchschnittliche Abweichungen über 3% treten nicht auf. Die höheren Abweichungen sind auch in der Berechnung der Massenströme im Fernwärmenetz sichtbar. Erklären lässt sich dies durch Gleichung (3.1), in welcher ohne die Rücklauftemperatur T_{RL} eine Information zur direkten Berechnung des Massenstroms fehlt.

$$\dot{Q} = \dot{m} \cdot c_p \cdot (T_{VL} - T_{RL}) \quad (3.1)$$

Tabelle 6.17.: Maximale absolute und relative Abweichungen der neuronalen Netze zu *SIR 3S*[®] für das aggregierte Netz 2 und zehn Simulationstage basierend auf Beispieltag 1 im Februar.

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	<i>MAPE</i> mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	<i>MAPE</i> ohne T_{RL} (%)
\dot{m} gesamt (t/h)	1,579	0,083	0,015	80,846	3,299	1,092
\dot{m} HKW (t/h)	0,934	0,058	0,010	61,836	3,374	1,113
\dot{m} HW (t/h)	0,636	0,215	0,029	28,080	3,259	1,033
\dot{m} DEA (t/h)	1,192	0,164	0,021	32,312	3,258	1,116
v HW (m/s)	0,001	0,225	0,032	0,022	3,298	1,027
v DEA (m/s)	0,001	0,157	0,022	0,033	3,347	1,149
v HKW (m/s)	0,001	0,060	0,014	0,064	3,379	1,115
Min, Δp (bar)	0,021	0,739	0,139	0,110	4,091	0,702
Min, p_{RL} (bar)	0,014	0,532	0,116	0,104	4,775	0,831
Min, T_{VL} (°C)	71,926	163,574	0,585	71,826	163,347	0,616
P_{Pumpe} VL HKW (kW)	1,049	0,446	0,082	23,252	6,983	2,098
P_{Pumpe} RL HKW (kW)	0,863	0,444	0,074	19,953	7,667	1,998
P_{Pumpe} RL DEA (kW)	0,016	inf	inf	0,025	inf	inf
n_{Pumpe} VL HKW (min^{-1})	175,417	17,450	0,094	167,989	16,592	0,729
n_{Pumpe} RL DEA (min^{-1})	0,065	inf	inf	0,225	inf	inf

Die Ergebnisse für den Gesamtmassenstrom und den minimalen Differenzdruck an Beispieltag 1 sind in den Abbildungen 6.14a und 6.14b dargestellt. Auch hier ist ein geringer Unterschied zwischen den Ergebnissen der neuronalen Netze mit und ohne Rücklauftemperatur erkennbar. Während das Netz mit der Temperatur die *SIR 3S*[®] Ergebnisse fast genau trifft, entstehen durch deren Auslassen Abweichungen. Für die minimale Vorlauftemperatur sowie einige Pumpenleistungen und -drehzahlen sind auch für das neuronale Netz mit Rücklauftemperatur hohe relative Abweichungen vorhanden. Die Verläufe der Leistung der Vorlaufpumpe des HKW und der minimalen Vorlauftemperatur zeigen die Abbildungen 6.14c und 6.14d. Dort sind auch die Ausreißer in *SIR 3S*[®] sichtbar.

Tabelle 6.18 zeigt die Abweichungen größer 3% für den Julitag. Dies ist für 11 der 26 Ausgänge des neuronalen Netzes mit Rücklauftemperatur der Fall und für 16 der 26 Ausgänge des Netzes ohne den Temperatureingang. Hier treten auch für einige Ausgangsgrößen durchschnittliche relative Abweichungen größer 3% auf, welche rot markiert sind. Einige Abweichungen sind grau hinterlegt. Zum Beispiel erklären sich Abweichungen der Pumpenleistung und -drehzahl der DEA wieder durch den Aus-Zustand der Pumpe und sonst geringen absoluten Abweichungen. Die Abweichungen der minimalen Vorlauftemperatur entstehen hauptsächlich durch Ausreißer und Rechenfehler. Ansonsten weichen die Massenströme und Strömungsgeschwindigkeiten an der DEA und am HW sowie der gesamte Wärmeverbrauch und Wärmeverlust stärker ab. Absolut sind alle Werte klein, weshalb auch geringe absolute Abweichungen wieder große relative Abweichungen verursachen.

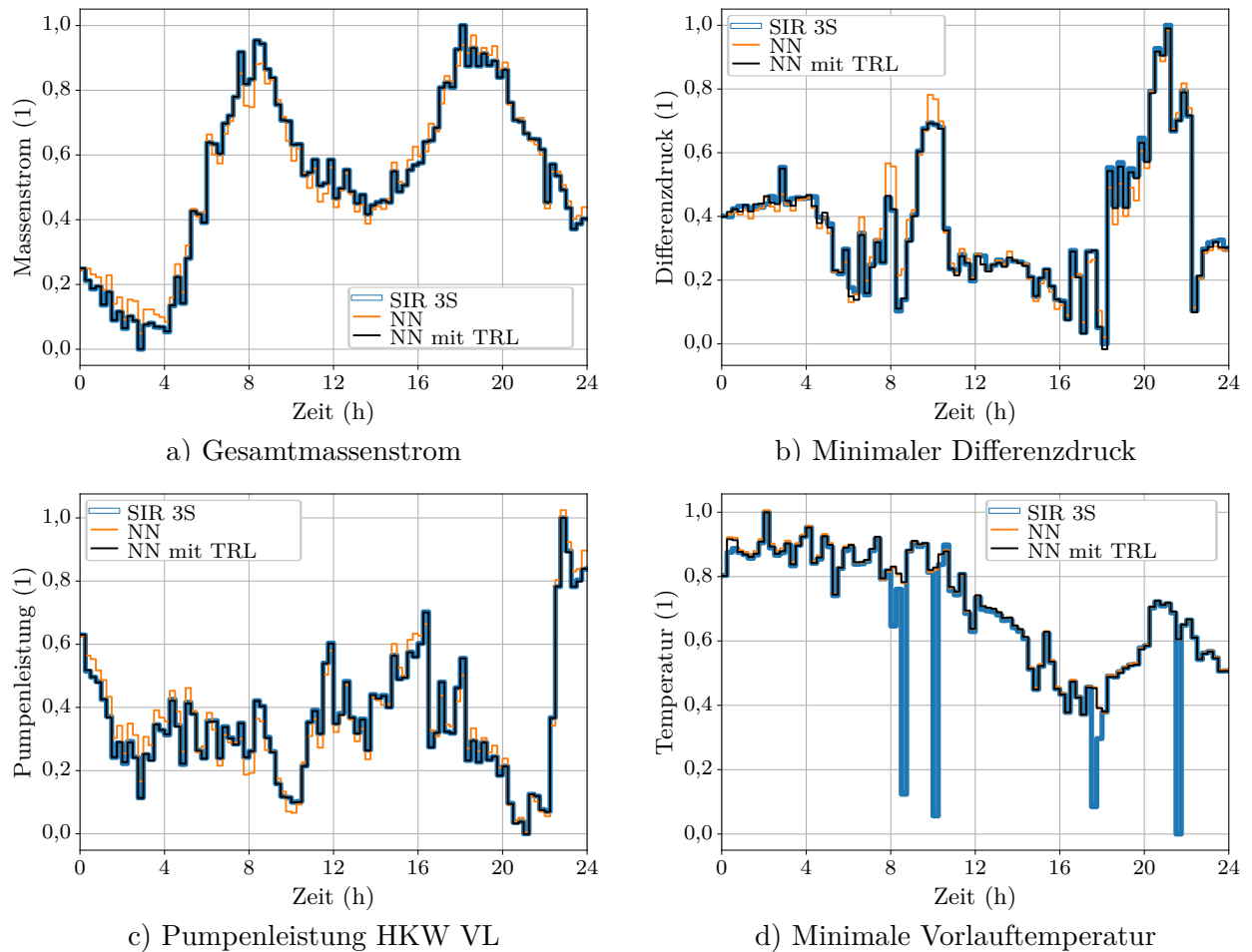


Abbildung 6.14.: Vergleich normierter Simulationsergebnisse von Netz 2 an Tag 1 im Februar mit dem *SIR 3S*[®] Modell, dem neuronalen Netz ohne (NN) und mit Rücklauftemperatur (NN mit TRL) als Eingangsvariable.

Tabelle 6.18.: Maximale absolute und relative Abweichungen der neuronalen Netze zu *SIR 3S*[®] für das aggregierte Netz 2 für zehn Simulationstage basierend auf Beispieltag 3 im Juli.

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	<i>MAPE</i> mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	<i>MAPE</i> ohne T_{RL} (%)
Wärmeverbrauch (MW)	1,277	40,815	11,180	1,331	41,346	13,339
\dot{m} gesamt (t/h)	1,678	0,665	0,240	26,206	8,302	1,937
\dot{m} HKW (t/h)	3,286	1,982	0,216	23,055	7,852	1,787
\dot{m} HW (t/h)	2,789	13,472	2,637	4,918	23,032	5,543
\dot{m} DEA (t/h)	38,919	49,184	15,043	39,338	58,115	16,437
v HW (m/s)	0,002	14,365	2,971	0,004	24,100	5,843
v DEA (m/s)	0,039	49,130	15,004	0,039	58,012	16,425
v HKW (m/s)	0,004	2,082	0,231	0,023	7,839	1,784
Min. p_{RL} (bar)	0,094	5,173	0,545	0,098	5,391	0,647
Min. T_{VL} (°C)	46,284	462,721	34,093	47,296	472,942	35,236
Wärmeverlust (MW)	0,250	4,196	1,779	0,258	4,445	1,050
P_{Pumpe} VL HKW (kW)	0,691	2,385	0,534	1,988	5,818	1,151
P_{Pumpe} RL HKW (kW)	0,533	1,246	0,208	1,524	3,752	1,097
P_{Pumpe} RL HW (kW)	1,741	15,189	3,420	1,787	15,585	3,103

Tabelle 6.18 fortgesetzt.

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	$MAPE$ mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	$MAPE$ ohne T_{RL} (%)
P_{Pumpe} RL DEA (kW)	0,019	inf	inf	0,038	inf	inf
n_{Pumpe} RL DEA (min^{-1})	0,074	inf	inf	0,756	inf	inf

Die Abbildungen 6.15a und 6.15b zeigen den Gesamtmassenstrom und den Wärmeverbrauch an dem Julitag. Die Massenströme und Strömungsgeschwindigkeiten an der DEA sind anschließend in den Abbildungen 6.15c und 6.15d dargestellt. Hier ist vor allem für die Massenströme und Strömungsgeschwindigkeiten zu erkennen, dass der Verlauf der Größen schlechter getroffen wird als zuvor an dem Beispieltag im Winter. In der späteren Optimierung werden die Massenströme der DEA nicht benötigt. Daher sind diese Ergebnisse nicht als kritisch zu beurteilen.

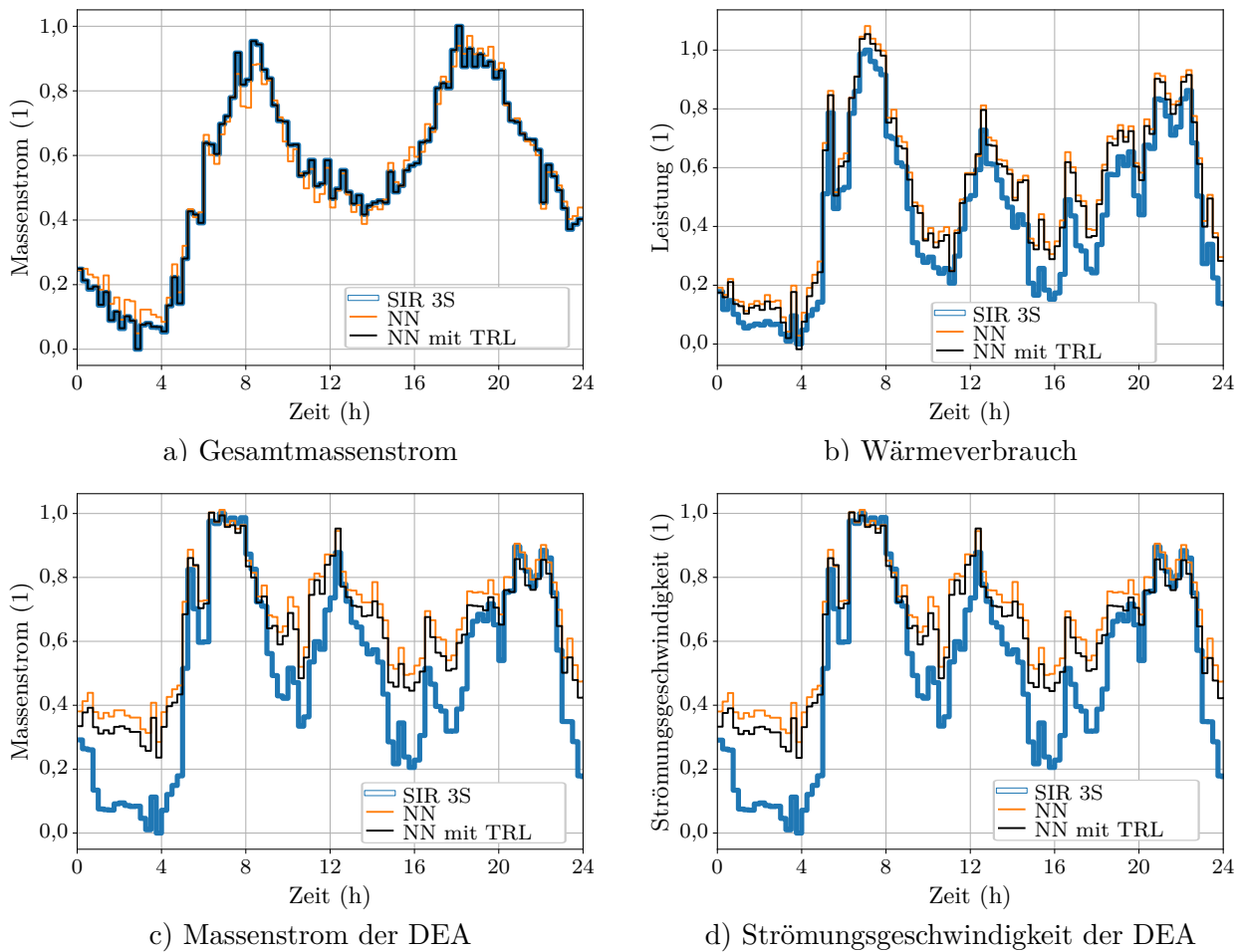


Abbildung 6.15.: Vergleich normierter Simulationsergebnisse von Netz 2 an Tag 3 im Juli mit dem $SIR\ 3S^{\circledR}$ Modell, dem neuronalen Netz ohne (NN) und mit Rücklaufftemperatur (NN mit TRL) als Eingangsvariable.

Insgesamt können mit neuronalen Netzen auch für Fernwärmenetz 2 gute Ergebnisse erzielt werden. Die Simulation des Wintertages kann mit einem neuronalen Netz mit der Rücklaufftemperatur als Eingangsvariable fast exakt nachgebildet werden. An wärmeren Tagen im Mai und Juli mit geringeren Massenströmen nimmt die Genauigkeit ab und die neuronalen Netze könnten durch weitere Trainingsdaten verbessert werden. Ein Großteil der benötigten Ausgangsgrößen kann aber auch hier von den

neuronalen Netzen sehr gut nachgebildet werden. Über alle Testdaten und die Variation der Beispieltage erzielt das neuronale Netz mit Rücklaufftemperatur eine durchschnittliche Abweichung unter 1 % verglichen mit *SIR 3S*[®] für alle in der Optimierung benötigten Ausgangsgrößen. Zu beachten ist, dass vor allem an Sommertagen zeitweise höhere maximale relative Abweichungen auftreten können.

Mit der Angabe der Rücklaufftemperatur kann die Vorhersage des neuronalen Netzes insgesamt deutlich verbessert werden, da es direkt aus den Leistungs- und Temperaturvorgaben auf die Massenströme im Netz schließen kann (siehe Gleichung (3.1)). Problematisch ist, dass Rücklaufftemperaturen zwar für Nachrechnungen von Beispieltagen bekannt sind, für Prognoserechnungen allerdings nicht. Somit müsste eine spätere Optimierung entweder basierend auf dem Modell ohne Rücklaufftemperatur oder inklusive ihrer Prognose durchgeführt werden.

6.2.3. Netz 2: Vorhersage der Rücklaufftemperatur

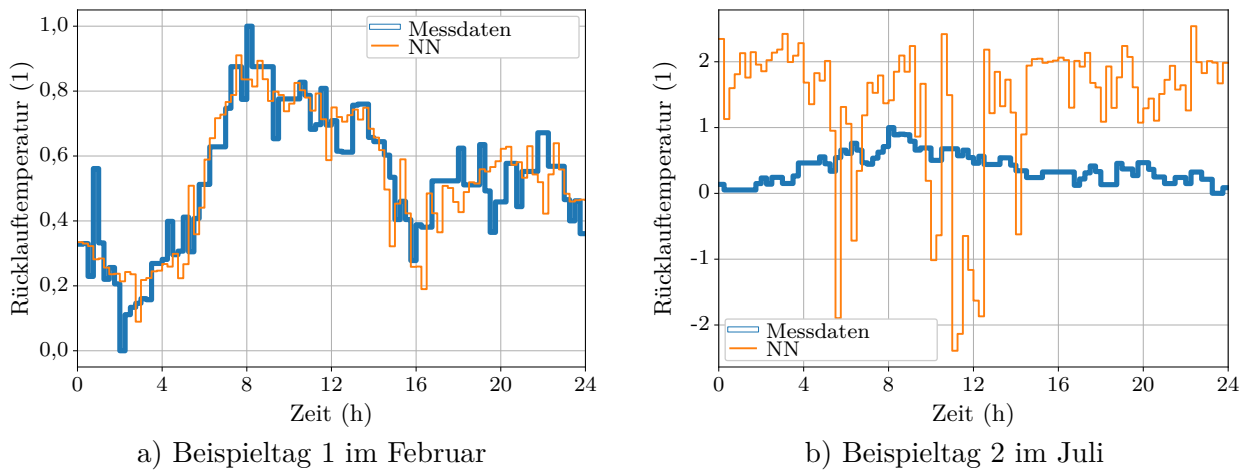
Im Anschluss wird deshalb die Möglichkeit untersucht, die Rücklaufftemperaturen des HKW basierend auf den Leistungen und Vorlaufftemperaturen des HKW und des HW vorherzusagen. Hier kann nicht direkt die vorab genannte Gleichung (3.1) verwendet werden, da bei einer Prognose nur die Leistung oder der Massenstrom fest vorgegeben werden können und somit einer der Werte nicht bekannt ist. Zudem beeinflussen die beiden Wärmeerzeugungsanlagen in Netz 2 auch gegenseitig ihre Rücklaufftemperaturen und die Flussrichtung über die DEA. Als Eingangsvariablen für eine Rücklaufftemperaturprognose werden vier aus der wirtschaftlichen Optimierung hervorgehende Sollwerte verwendet: Die Gesamtleistung, die Leistung des HKW sowie die Vorlaufftemperaturen des HKW und des HW.

Nachfolgend werden Messdaten von 19 Wintertagen mit insgesamt 27 360 Datensätzen verwendet. Dynamische Effekte der Temperaturausbreitung und deren Einfluss auf die Rücklaufftemperaturen werden nicht beachtet. Pro Zeitpunkt wird den vier Eingangsvariablen also eindeutig eine Rücklaufftemperatur zugeordnet. Die Daten werden für das Training in 70 % Trainingsdaten, 15 % Validierungsdaten und 15 % Testdaten geteilt und für das Training neuronaler Netze verwendet. Zum Einsatz kommt wieder die *Python* Bibliothek *PyTorch*. Die neuronalen Netze verwenden den Optimierer *Adam* mit einer Lernrate von 10^{-3} , die *Loss*-Funktion *MAE* und die Aktivierungsfunktion *Swish*. Die Auswahl einer Netzkonfiguration wird hier nicht im Detail präsentiert. Das ausgewählte Modell besteht aus zwei versteckten Schichten mit jeweils 32 Neuronen und wurde für 10 000 Epochen trainiert. Die Vorhersage der Rücklaufftemperatur zeigt insgesamt gute Ergebnisse basierend auf den 19 Beispieltagen. Tabelle 6.19 zeigt die maximalen und durchschnittlichen absoluten sowie relativen Abweichungen für die Testdaten. Der durchschnittliche relative Fehler (MAPE) liegt unter einem Prozent, die maximale relative Abweichung unter fünf Prozent.

Tabelle 6.19.: Abweichungen des neuronalen Netzes zu den Testdaten der HKW-Rücklaufftemperatur-Vorhersage in Fernwärmenetz 2.

	Max. abs. Fehler (°C)	MAE (°C)	Max. rel. Fehler (%)	MAPE (%)
T_{RL} HKW	2,679	0,279	4,563	0,491

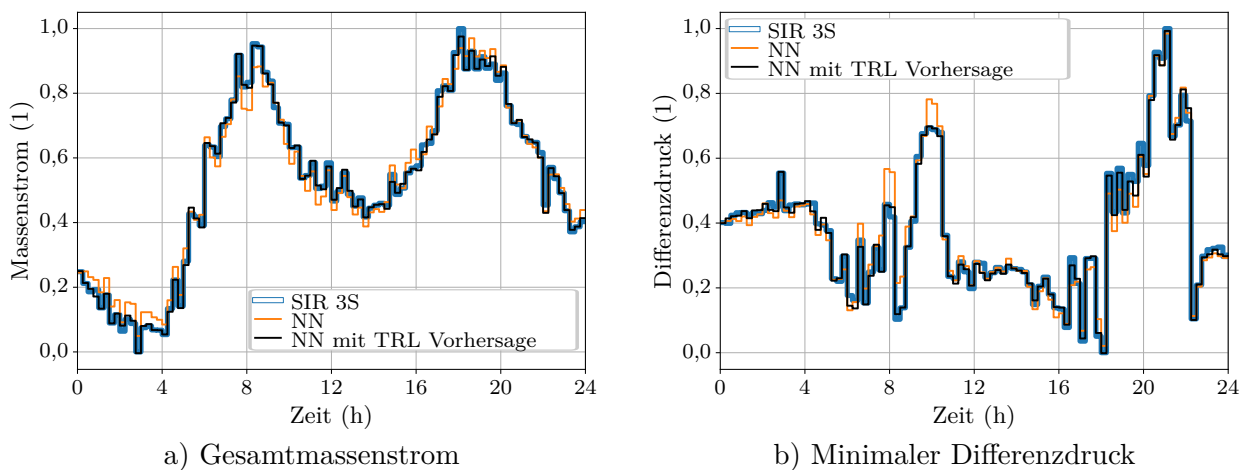
In Abbildung 6.16 werden die Rücklaufftemperaturvorhersagen für Beispieltag 1 und 3 mit Messdaten verglichen. Es ist ersichtlich, dass die Rücklaufftemperaturen an dem Wintertag sehr gut vorhergesagt werden können, während die Vorhersage für den Sommertag scheitert, da nur Wintertage für das Training des neuronalen Netzes verwendet wurden. Das neuronale Netz für die Rücklaufftemperaturvorhersage kann somit in dem jetzigen Status nur für Tage im Winter verwendet werden.



a) Beispieltag 1 im Februar

b) Beispieltag 2 im Juli

Abbildung 6.16.: Normierte Rücklaufftemperatur des HKW in Netz 2. Messdaten verglichen mit der Vorhersage durch ein Neuronales Netz (NN).



a) Gesamtmassenstrom

b) Minimaler Differenzdruck

Abbildung 6.17.: Vergleich normierter Simulationsergebnisse von Netz 2 an Tag 1 im Februar mit dem *SIR 3S*[®] Modell, dem neuronalen Netz ohne (NN) und mit Vorhersage der Rücklaufftemperatur (NN mit TRL Vorhersage).

In Abbildung 6.17 wird zudem die für Tag 1 berechnete Rücklaufftemperatur für eine Simulation mit dem neuronalen Netz für Fernwärmenetz 2 verwendet (schwarz) und mit einer Berechnung ohne die Rücklaufftemperatur (orange) und in *SIR 3S*[®] (blau) verglichen. Dargestellt sind der gesamte Massenstrom im Netz in Abbildung 6.17a und der minimale Differenzdruck in Abbildung 6.17b. Hier wird deutlich, dass mit der vorhergesagten Rücklaufftemperatur die Ergebnisse von *SIR 3S*[®] für beide Größen nahezu exakt getroffen werden und verglichen mit einer Berechnung ohne die Rücklaufftemperatur deutlich genauere Simulationen möglich sind.

Zusammenfassend wurde gezeigt, dass eine Berechnung der Rücklaufftemperatur basierend auf Erzeugungsleistungen und Vorlaufftemperaturen möglich ist und Simulationsergebnisse deutlich verbessern kann. Allerdings basiert die Berechnung auf Messdaten. Somit kann eine genaue Prognose der Rücklaufftemperaturen bei Änderungen am Netz, zum Beispiel durch bauliche Maßnahmen oder Erweiterung des Netzes um neue Erzeugungsanlagen, nicht garantiert werden. Zudem sind Messdaten aus allen Jahreszeiten notwendig, um das gesamte Netzverhalten zu erlernen. Bisher wurden nur Messdaten von Wintertagen verwendet, weshalb auch nur für Wintertage eine gute Rücklaufftemperaturprognose möglich ist.

6.3. Fernwärmenetz 3

Im nächsten Schritt wird Fernwärmenetz 3 entsprechend Abbildung 6.18 modelliert. Das Netz besteht aus zwei Teilgebieten, welche durch eine DEA miteinander verbunden sind. Die DEA kann über Pumpen im Vor- und Rücklauf bidirektional Wärme verschieben. Die Wärme wird überwiegend in Teilgebiet (TG) 1 von HKW 1 inklusive einer Geothermieanlage und HW 1 bereitgestellt. Des Weiteren stehen HW 2 sowie im Teilgebiet 2 HW 3 mit zusätzlicher Geothermieanlage zur Verfügung. Fernwärmenetz 3 ist das größte in dieser Arbeit behandelte Netz und weist mit vier Wärmeerzeugungsanlagen zudem die höchste Komplexität auf. Durch zwei Geothermieanlagen ergeben sich zusätzliche in der Optimierung in Kapitel 7.3 zu berücksichtigende Restriktionen. In Kapitel 6.3.1 wird zunächst wieder ein thermo-hydraulisches Netzmodell vorgestellt. Basierend auf den Ergebnissen für die Netze 1 und 2 werden anschließend in Kapitel 6.3.2 neuronale Netze für die Modellvereinfachung verwendet und ihre Anwendung auf ein komplexes Fernwärmenetz mit verteilter regenerativer Wärmeeinspeisung getestet.

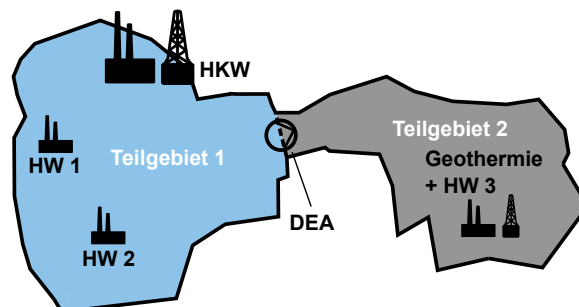


Abbildung 6.18.: Vereinfachte, nicht-maßstäbliche Darstellung des realen Fernwärmenetzes 3 mit Druckerhöhungs- und Wärmeerzeugungsanlagen.

6.3.1. Netz 3: *SIR 3S*[®]

Für Netz 3 ist ein vollständiges Netzmodell mit etwa 31 000 Knoten und Rohrleitungen verfügbar. Eine aggregierte Version des Netzes wird nicht erstellt und Wärmeerzeugungsanlagen sind nur vereinfacht als einzelne Anlagen mit Umwälzpumpen modelliert. Das *SIR 3S*[®] Modell wurde von der 3S Consult GmbH mit den gleichen drei Beispieltagen wie Netz 2 validiert und hat folgende Eingangsvariablen:

- Gesamtmassenströme der Teilgebiete 1 und 2 und der drei Heizwerke
- Vorlauftemperaturen des HKW und der HWs
- Rücklauftemperaturen der Teilgebiete
- Druckhaltung des HKW und Soll-Differenzdrücke beider Teilgebiete
- Betriebsmodus der DEA oder Vorgabe einer Soll-Drehzahl. Fünf relevante Betriebsmodi sind in dem *SIR 3S*[®] Modell integriert. An den Beispieltagen traten keine anderen Betriebsmodi auf.

Der Massenstrom des HKW ergibt sich aus der Differenzdruckregelung der Pumpen. Die Betriebsweise des Netzes wird so modelliert, dass Druckmesswerte gut mit den Modellergebnissen übereinstimmen. Eine Temperaturkalibrierung wird hier nicht durchgeführt. Die Schlechtpunkte der Druckregelung entsprechen jenen in der Realität und auch die Betriebsarten der DEA wurden diesen angepasst. Allerdings fehlen noch einige Druckregelpunkte in dem Modell, weshalb Prognoserechnungen die Realität nicht vollständig abbilden können. Dennoch ergibt sich bei Nachberechnungen bereits eine gute Übereinstimmung des *SIR 3S*[®] Modells mit Messdaten. Tabelle 6.20 zeigt die mittleren 24 Stunden Druck- und Temperaturabweichungen. Aufgrund der fehlenden Temperaturkalibrierung sind die

Temperaturabweichungen höher als zum Beispiel für Netz 2 in Tabelle 6.10. Eine quasi-stationäre 24-Stunden-Simulation mit 15-Minuten-Zeitschritten dauert mit dem Modell wenige Minuten.

Bei einer Inbetriebnahme des Regelungskonzepts muss die Druckregelung des Modells vervollständigt und eine Temperaturkalibrierung durch Anpassung der Wärmedurchgangskoeffizienten sowie der Rücklauftemperaturverteilung durchgeführt werden. Die Druckregelung kann dabei auch teilweise von einer nachgelagerten Optimierung übernommen werden.

Tabelle 6.20.: Mittlere 24h-Abweichungen der Nachberechnung von Fernwärmenetz 3 in *SIR 3S*[®] an den drei Beispieltagen.

	Tag 1	Tag 2	Tag 3
Druck (bar)	0,21	0,20	0,27
Standardabweichung Druck (bar)	0,17	0,17	0,42
Druck (%)	3,5	3,3	6,0
Temperatur (°C)	7,21	7,16	4,85
Standardabweichung Temperatur (°C)	13,52	10,12	8,31
Temperatur (%)	10,0	9,5	7,7

6.3.2. Netz 3: Neuronale Netze

Aufgrund der hohen Rechenzeiten von thermo-hydraulischen Modellen ist auch für Fernwärmenetz 3 eine Modellvereinfachung notwendig. Für Netz 1 und 2 haben sich neuronale Netze als beste Option für die Approximation stationärer *SIR 3S*[®] Simulationen herausgestellt. Hier soll dieser Ansatz nun auf ein komplexes Fernwärmenetze mit verteilter regenerativer Wärmeeinspeisung angewandt werden.

So werden zunächst 2 909 903 stationäre Simulationen mit variierten Eingangsgrößen des *SIR 3S*[®] Netzmodells ausgeführt, um anschließend neuronale Netze zu trainieren. Die Eingangsgrößen werden wie für Netz 2 zwischen den Minimal- und Maximalwerten der drei Beispieltage variiert. Massenströme und Vorlauftemperaturen werden zusätzlich etwas über diesen Bereich hinaus abgesenkt oder angehoben, um auch potenzielle Situationen mit niedrigeren oder höheren Wärmeeinspeisungen zu berücksichtigen. Die DEA wird je nach Eingangsdatensatz in den fünf Betriebsarten simuliert. Die Pumpen der DEA haben zudem eine Mindestdrehzahl, welche nicht direkt in *SIR 3S*[®] berücksichtigt wird. Simulationen mit eingeschalteter DEA und niedrigerer resultierender Drehzahl der Pumpen werden deshalb mit Vorgabe der Minimaldrehzahl wiederholt. Die Rücklauftemperaturen und Druckhaltung des HKW werden wie an den Beispieltagen gemessen simuliert. Insgesamt kann mit den verwendeten Eingangsdaten der gesamte Betriebsbereich der drei Beispieltage abgedeckt werden. Nach dem Aus-sortieren fehlerhafter Simulationsergebnisse (Pumpenleistungen, Massenströme und Drücke kleiner 0) bleiben für das Training 1 985 315 Datensätze übrig. Anzumerken ist, dass ein großer Teil dieser fehlerhaften Ergebnisse durch die Wahl der Eingangsdaten über den Betriebsbereich des Netzes hinaus entsteht. Auch die Kombination aus Leistungen der Wärmeerzeugungsanlagen und Betriebsart der DEA kann zu nicht plausiblen Ergebnissen führen.

Für das Training von Modellen werden die Gesamtleistungen der zwei Teilgebiete und Leistungen der einzelnen HWs statt deren Massenströmen verwendet. So können die Sollwerte der wirtschaftlichen Optimierung direkt in ein Modell gegeben werden. Die Massenströme sind anschließend ein Ausgang der Modelle. Die folgenden 17 Eingänge werden für die zu trainierenden Modelle verwendet:

- Gesamte Wärmeleistung und relativer Wärmeverbrauch von Teilgebiet 2
- Leistungen und Vorlauftemperaturen der drei HWs sowie die Vorlauftemperatur des HKW
- Zielwert der Druckhaltung

- Sechs *Booleans* für die Betriebsarten der DEA sowie der Drehzahlregelung. Es wurde auch untersucht jeder Betriebsart einen *Integer* zuzuordnen. Die Ergebnisse der trainierten neuronalen Netze sind ähnlich.
- Rücklauftemperaturen der Teilgebiete

Als Ausgänge der Modelle werden 43 Größen trainiert, welche später in Optimierungen als Zielgrößen oder Nebenbedingungen (Restriktionen) zum Einsatz kommen:

- Leistung des HKW
- Massenströme des HKW, der drei HWs, der DEA und der zwei Teilgebiete
- Pumpendrehzahlen und -leistungen der Pumpen des HKW, der HWs und der DEA
- Strömungsgeschwindigkeiten an den Wärmeerzeugungsanlagen und der DEA
- Minimale Differenzdrücke an den Schlechtpunkten der zwei Teilgebiete
- Minimale und maximale Vor- und Rücklaufdrücke an den Schlechtpunkten der zwei Teilgebiete
- Minimale Vorlauftemperatur der beiden Teilgebiete
- Wärmeverluste und -verbräuche
- Fehlmassenstrom. Dieser gibt den fehlenden Massenstrom an, um die Massenstrombilanz im System zu erfüllen, falls die Regelung in *SIR 3S*[®] die Sollwerte nicht genau erreicht.
- Vor- und Rücklaufdrücke am HKW und HW 3

Untersuchung der Anzahl der Schichten und Neuronen sowie der Lernrate

Mit den Simulationsdaten soll nun ein neuronales Netz trainiert werden, welches *SIR 3S*[®] Simulationen möglichst gut abbildet. Dafür werden, basierend auf den Ergebnissen für Fernwärmenetz 1 und 2, neuronale Netze mit der Bibliothek *PyTorch*, ein bis vier Schichten und unterschiedlicher Anzahl an Neuronen getestet. Für die Auswahl werden Modelle mit der Rücklauftemperatur als Eingang trainiert. Tabelle A.15 im Anhang zeigt die Trainingsergebnisse der neuronalen Netze nach 100 Epochen. Die restlichen Einstellungen der neuronalen Netze bleiben wie bei Netz 1 und 2. Es wird also der Optimierer *Adam* mit einer Lernrate von 10^{-3} verwendet, die *Loss*-Funktion MAE, die Aktivierungsfunktion *Swish*, die Gewichte werden mit *He-Uniform* initialisiert und die Daten zufällig in 60 % Trainings-, 20 % Validierungs- und 20 % Testdaten unterteilt. Es sind somit 1 191 189 Datensätze für das Training und jeweils 397 063 Datensätze für Validierung und Test vorhanden, welche zwischen 0 und 1 normiert werden. Im Anhang erfolgt eine Bewertung der unterschiedlichen Netze anhand der Werte MAE, MSE, RMSE und R^2 sowie einiger spezifischer Ergebnisse. Dazu gehören der MAPE des minimalen Differenzdrucks in Teilgebiet 2, der MAE des maximalen Rücklaufdrucks in Teilgebiet 1 sowie des Massenstroms des HKW. Die besten Modelle werden für weitere Untersuchungen ausgewählt.

Die Lernrate soll für die neuronalen Netze für Fernwärmenetz 3 weiter getestet werden. Für Netz 1 zeigte ein neuronales Netz ohne Reduzierung der Lernrate die besten Ergebnisse, während für Netz 2 mit Reduzierung eine bessere Approximation der *SIR 3S*[®] Simulationen möglich war. Die im Anhang nach einer Bewertung ausgewählten Netzkonfigurationen werden so als Vergleich für 1000 Epochen mit konstanter Lernrate und mit einer Halbierung der Lernrate alle 200 Epochen (mit *Decay*) trainiert. Die Ergebnisse zeigt Tabelle 6.21. Aufgelistet werden der durchschnittliche MAPE und MAE für die Druckausgänge der Netze. Für den durchschnittlichen „MAPE gesamt“ werden 17 der 43 Ausgangsgrößen der neuronalen Netze verwendet. Die anderen Werte, wie zum Beispiel die Leistungen, Massenströme und Strömungsgeschwindigkeiten an den HWs oder der DEA, schwanken häufig um 0, weshalb selbst bei geringen Abweichungen hohe prozentuale Fehler auftreten können. Diese Werte

werden folglich in der Berechnung des „MAPE gesamt“ nicht berücksichtigt. Auch Ausgänge, welche später nicht mehr für die Optimierung benötigt werden, fließen nicht in den durchschnittlichen Fehler ein. Eine Beschreibung der Ausgangsgrößen und Begründung, warum einige nicht für die Berechnung der Modellgenauigkeit verwendet werden, kann im Anhang in Tabelle A.17 nachgelesen werden. Eine vollständige Auswertung aller Ausgangsgrößen folgt zudem später mit dem final ausgewählten neuronalen Netz. Die in der Tabelle mit t_s bezeichnete durchschnittliche Rechenzeit der Modelle wird aus 100 in Schleife durchgeführten 24-Stunden-Simulationen im 15-Minuten-Takt berechnet. Verwendet wird ein Rechner mit *Intel Core i7-4771* (4 x 3,50 GHz) und 32 GB RAM.

Tabelle 6.21.: Bewertung neuronaler Netze mit Rücklauftemperatur mit Testdaten für Fernwärmenetz 3 nach 1000 Epochen Training.

Neuronen pro Schicht	<i>MAPE</i> Drücke (%)	<i>MAE</i> Drücke (bar)	<i>MAPE</i> gesamt (%)	R^2	t_s (ms)
129	1,823	0,075	1,626	0,95884	0,086
258/172	0,593	0,024	0,578	0,98773	0,158
215/172/86	0,377	0,016	0,390	0,98056	0,181
215/172/172	0,354	0,015	0,374	0,99006	0,193
258/258/172/43	0,302	0,013	0,326	0,98509	0,248
1548/774/516/774	0,324	0,015	0,344	0,98224	2,086
1634/1118/860/860	1,578	0,066	1,426	0,93875	3,131
129 (mit <i>Decay</i>)	1,704	0,073	1,535	0,96258	0,084
258/172 (mit <i>Decay</i>)	0,390	0,016	0,398	0,98767	0,147
215/172/86 (mit <i>Decay</i>)	0,220	0,009	0,248	0,98938	0,186
215/172/172 (mit <i>Decay</i>)	0,204	0,008	0,232	0,98183	0,190
258/258/172/43 (mit <i>Decay</i>)	0,168	0,007	0,201	0,98488	0,256
1548/774/516/774 (mit <i>Decay</i>)	0,135	0,005	0,167	0,98507	2,162
1634/1118/860/860 (mit <i>Decay</i>)	0,099	0,004	0,128	0,95783	3,101

Einige der neuronalen Netze in Tabelle 6.21 werden bereits aussortiert und sind rot markiert. Dazu gehören die Netze mit einem MAPE größer 1% und Simulationszeiten größer 2ms. Im Vergleich schneiden andere Modelle für diese Größen deutlich besser ab. Vor allem die Simulationszeiten der großen Netze mit 4 Schichten und einer hohen Anzahl an Neuronen sind deutlich höher als die der anderen Netze. Für die beiden größten Netze mit 1634, 1118, 860 und 860 Neuronen pro Schicht ist sogar eine Abnahme der Genauigkeit bezüglich der Testdaten bzw. des R^2 -Werts zu erkennen. Dies weist auf eine Überanpassung (*Overfitting*) der großen Netze hin.

Für eine finale Bewertung werden die Modelle in Tabelle 6.22 auf den Winter-Beispieltag 1 und in Tabelle 6.23 auf den Sommertag 3 angewandt. Wie für Netz 2 gehen für Netz 3 die Rücklauftemperaturen der Beispieltage in die Trainings- und Testdaten ein. Für eine unabhängige Bewertung werden pro Beispieltag zehn 24-Stunden-Simulationen mit Variation der Eingangswerte in *SIR 3S*[®] durchgeführt. Massenströme werden zufällig um bis zu $\pm 5\%$, die Druckhaltung um bis zu $\pm 2\%$, Vorlauftemperaturen um bis zu $\pm 3^\circ\text{C}$ und Rücklauftemperaturen um bis zu $\pm 1^\circ\text{C}$ verändert. Für den Sommertag findet nur eine Variation in die positive Richtung statt, da bei Verringerung der Werte teils negative Drücke in *SIR 3S*[®] auftreten. Hier wird zusätzlich die maximale relative Abweichung der entsprechend Tabelle A.17 relevanten Ausgangsgrößen für die Bewertung verwendet.

Tabelle 6.22.: Abweichung verschiedener neuronaler Netze für Fernwärmenetz 3 zu *SIR 3S*[®] für zehn Simulationstage basierend auf Tag 1 im Februar.

Neuronen pro Schicht	<i>MAPE</i> Drücke (%)	<i>MAE</i> Drücke (bar)	<i>MAPE</i> gesamt (%)	Max. rel. Error (%)
258/172	0,399	0,019	0,298	11,644
215/172/86	0,278	0,012	0,217	7,186
215/172/172	0,226	0,010	0,206	8,937
258/258/172/43	0,203	0,010	0,187	10,098
258/172 (mit <i>Decay</i>)	0,318	0,015	0,204	9,834
215/172/86 (mit <i>Decay</i>)	0,135	0,006	0,099	7,027
215/172/172 (mit <i>Decay</i>)	0,139	0,006	0,103	6,883
258/258/172/43 (mit <i>Decay</i>)	0,081	0,004	0,073	6,983

Tabelle 6.23.: Abweichung verschiedener neuronaler Netze für Fernwärmenetz 3 zu *SIR 3S*[®] für zehn Simulationstage basierend auf Tag 3 im Juli.

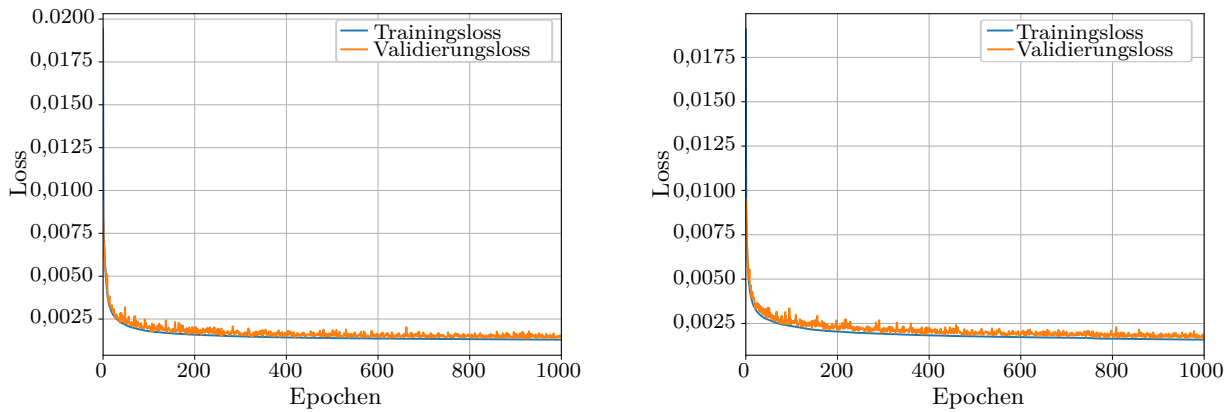
Neuronen pro Schicht	<i>MAPE</i> Drücke (%)	<i>MAE</i> Drücke (bar)	<i>MAPE</i> gesamt (%)	Max. rel. Error (%)
258/172	0,806	0,029	0,946	19,702
215/172/86	0,873	0,032	1,139	22,020
215/172/172	0,705	0,026	0,830	16,996
258/258/172/43	0,856	0,029	0,821	28,219
258/172 (mit <i>Decay</i>)	0,894	0,033	0,867	27,940
215/172/86 (mit <i>Decay</i>)	0,810	0,029	0,710	22,634
215/172/172 (mit <i>Decay</i>)	0,769	0,027	0,714	22,989
258/258/172/43 (mit <i>Decay</i>)	0,649	0,023	0,644	21,709

Auch für die zwei Beispieltage ist kein deutlicher Unterschied zu erkennen. Allerdings wird später festgestellt, dass eine Optimierung der Beispieltage nicht mit allen neuronalen Netzen möglich ist. Daher wird in der folgenden Tabelle 6.24 die Optimierung der Beispieltage mit den neuronalen Netzen untersucht. Für die Netze ist bei erfolgreicher Optimierung jeweils die Optimierungsdauer in Sekunden angegeben. Nicht erfolgreiche Optimierungen sind mit einem X markiert. Die durchgeführten Optimierungen werden im weiteren Verlauf in Kapitel 7.1.2 erläutert.

Tabelle 6.24.: Optimierungsdauer der neuronalen Netze für Fernwärmenetz 3 an den drei Beispieltagen.

Neuronen pro Schicht	Tag 1 (Februar) (s)	Tag 2 (Mai) (s)	Tag 3 (Juli) (s)
258/172	17,44	X	X
215/172/86	16,24	X	X
215/172/172	15,97	65,39	322,14
258/258/172/43	14,23	27,99	30,61
258/172 (mit <i>Decay</i>)	17,43	X	X
215/172/86 (mit <i>Decay</i>)	21,52	X	X
215/172/172 (mit <i>Decay</i>)	20,52	X	X
258/258/172/43 (mit <i>Decay</i>)	19,01	X	X

Die Optimierung für Beispieltag 1 ist mit allen neuronalen Netzen erfolgreich, die für Tag 2 und 3 scheitern allerdings mit den meisten Netzen. Der Grund sind unterschiedliche Restriktionen, für welche die Netze Ausreißer außerhalb des gültigen Bereichs vorhersagen und die Optimierung so keine Lösung



a) NN mit Rücklaufftemperatureingang

b) NN ohne Rücklaufftemperatureingang

Abbildung 6.19.: Lernkurven für die neuronalen Netze (NN) von Fernwärmenetz 3.

findet. Nur mit zwei der Netze kann das Optimierungsproblem für alle drei Tage gelöst werden, wobei die Berechnung mit dem ersten Netz deutlich länger dauert. Somit wird das neuronale Netz mit vier versteckten Schichten und 258, 258, 172 und 43 Neuronen trainiert mit konstanter Lernrate für eine detaillierte Ergebnisauswertung im folgenden Abschnitt und für die Optimierung in Kapitel 7.3 gewählt.

Neuronale Netze für Fernwärmenetz 3

Das Netz mit 258, 258, 172 und 43 Neuronen wird zusätzlich ohne die Rücklaufftemperaturen als Eingang trainiert und zeigt dabei keine guten Berechnungsergebnisse für die Beispieltage 2 und 3. Stattdessen wird das Netz mit 215, 172 und 172 gewählt, welches bessere Berechnungsergebnisse liefert. Für die neuronalen Netze wurde wieder eine 5-fache Kreuzvalidierung durchgeführt, um sicherzugehen, dass keine Überanpassung an bestimmte Teilmengen der Lerndaten vorkommt. Alle fünf trainierten Modelle zeigen dabei ähnlich geringe Abweichungen zu den Validierungsdaten. Die Lernkurven der beiden final trainierten neuronalen Netze sind in Abbildung 6.19 dargestellt. Zusammenfassend zeigt die folgende Tabelle 6.25 die durchschnittlichen Abweichungen für Drücke und die entsprechend Tabelle A.17 relevanten Ausgangsgrößen für die Testdaten und drei Beispieltage.

Tabelle 6.25.: Durchschnittliche Abweichung der neuronalen Netze zu $SIR 3S^{\circledast}$ für Fernwärmenetz 3.

	MAPE Drücke (%)	MAE Drücke (bar)	MAPE gesamt (%)
Neuronales Netz mit Rücklaufftemperatur			
Testdaten	0,302	0,013	0,326
Zehn Simulationstage basierend auf Tag 1	0,203	0,010	0,187
Zehn Simulationstage basierend auf Tag 2	0,196	0,008	0,666
Zehn Simulationstage basierend auf Tag 3	0,856	0,029	0,821
Neuronales Netz ohne Rücklaufftemperatur			
Testdaten	0,412	0,019	0,458
Zehn Simulationstage basierend auf Tag 1	0,499	0,027	0,663
Zehn Simulationstage basierend auf Tag 2	0,482	0,024	1,685
Zehn Simulationstage basierend auf Tag 3	0,933	0,037	2,087

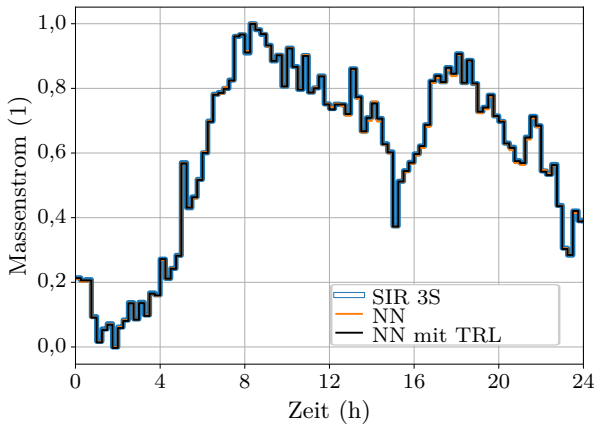
Die Tabellen A.18 bis A.20 im Anhang zeigen die maximalen absoluten und relativen Abweichungen aller Ausgangsgrößen verglichen mit $SIR 3S^{\circledast}$ für die neuronalen Netze mit und ohne Rücklaufftemperatur. Wie für die vorherigen Fernwärmenetze liefern die neuronalen Netze für den Wintertag die besten

Ergebnisse, während die Genauigkeit für die wärmeren Tage abnimmt. Hauptsächlich nehmen die relativen Abweichungen zu, während die absoluten Fehler für die wärmeren Beispieltage nur leicht steigen. In der folgenden Tabelle 6.26 werden alle maximalen Abweichungen größer 3 % der für die Optimierung benötigten Ausgangsgrößen für Beispieltag 1 und das neuronale Netz mit Rücklauftemperatur aufgelistet. Das Netz ohne Rücklauftemperatur weist für fast alle Ausgänge maximale Abweichungen größer 3 % auf. Die durchschnittliche Abweichung liegt niedriger. Für dieses neuronale Netz und die anderen Beispieltage wird die Auswertung nur im Anhang durchgeführt. Relative Abweichungen größer 3 % verglichen mit *SIR 3S*[®] treten an dem Tag im Februar für 14 der 43 Ausgangsgrößen auf. Für die grau markierten Größen treten die Abweichungen aufgrund von Werten nahe 0 auf. Geringe absolute Abweichungen verursachen bereits hohe relative Fehler. Für die minimale Vorlauftemperatur tritt wieder ein inzwischen behobener Berechnungsfehler im Interface zu *SIR 3S*[®] auf. Auch für den Massenstrom in Teilgebiet 2 und drei Druckausgänge treten relative Abweichungen über 3 % auf, die absoluten und durchschnittlichen relativen Abweichungen sind allerdings gering. Nicht gut vorhergesagt werden können die Drehzahlen der Vor- und Rücklaufpumpen der DEA und der Fehlmassenstrom. Für die DEA ist anzumerken, dass die Pumpen am Beispieltage ohne eine Drehzahlvorgabe in der Simulation mit *SIR 3S*[®] und neuronalen Netzen außerhalb ihres Drehzahlbereichs laufen. Der Vergleich wird ohne Drehzahlvorgabe durchgeführt, welche in den Modellen aber möglich ist.

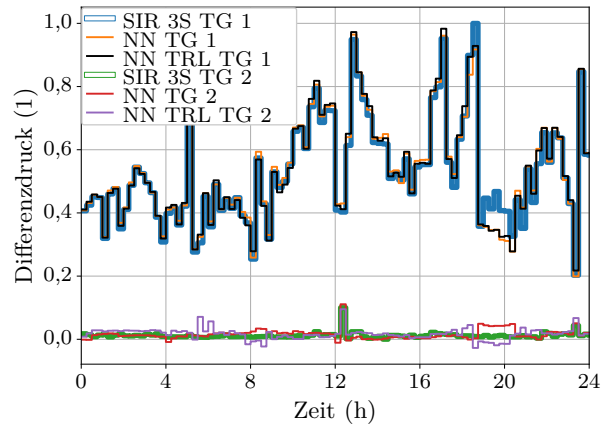
Tabelle 6.26.: Maximale absolute und relative Abweichungen des neuronalen Netzes mit Rücklauftemperatur zu *SIR 3S*[®] für Netz 3 und zehn Simulationstage basierend auf Tag 1.

Ausgangsgröße	Max. abs. Fehler	Max. rel. Fehler (%)	MAPE (%)
\dot{m} Teilgebiet 2 (t/h)	16,570	3,077	0,196
\dot{m} HW 2 (t/h)	0,586	inf	inf
\dot{m} DEA (t/h)	17,327	445,667	3,828
n_{Pumpe} RL HW 2 (min^{-1})	2,419	inf	inf
n_{Pumpe} VL DEA (min^{-1})	94,703	inf	inf
n_{Pumpe} RL DEA (min^{-1})	38,930	inf	inf
v HW 2 (m/s)	0,002	inf	inf
v DEA (m/s)	0,040	304,296	3,516
Min. Δp Teilgebiet 1 (bar)	0,236	10,098	0,398
Min. Δp Teilgebiet 2 (bar)	0,059	3,272	0,347
Min. p_{RL} Teilgebiet 2 (bar)	0,164	4,542	0,311
Min. T_{VL} Teilgebiet 1 ($^{\circ}\text{C}$)	6,999	6,926	0,300
Min. T_{VL} Teilgebiet 2 ($^{\circ}\text{C}$)	43,842	80,277	0,492
Fehlmassenstrom (t/h)	29,268	inf	126,160

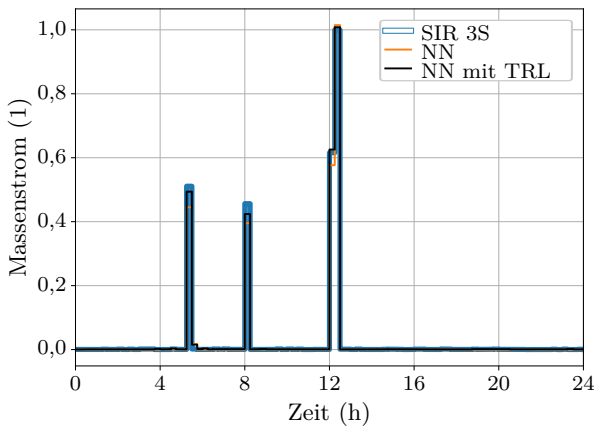
Abbildung 6.20 zeigt einige der Berechnungsergebnisse. Der Massenstrom des HKW und der minimale Differenzdruck in beiden Teilgebieten sind in den Abbildungen 6.20a und 6.20b dargestellt. Der Massenstrom kann sowohl von dem Modell mit als auch ohne Rücklauftemperatur nahezu exakt vorhergesagt werden. Die minimalen Differenzdrücke der beiden Teilgebiete werden ebenfalls mit guter Genauigkeit berechnet. Die Abbildungen 6.20c und 6.20d zeigen die Berechnung des Fehlmassenstroms und der minimalen Vorlauftemperatur. Der Fehlmassenstrom wird als Ergebnisgröße des *SIR 3S*[®] Modells ausgegeben, falls zu einem Zeitpunkt die Massenstrombilanz im Netz von der Druckregelung nicht erfüllt werden kann. Solche Zustände können durch die komplexe Regelung der DEA auftreten und werden ab einem Fehlmassenstrom von 10 t/h als kritisch bewertet. Maximale Abweichungen von bis zu 30 t/h für die neuronalen Netze mit und ohne Rücklauftemperatur sind also kein gutes Ergebnis. Die hohen Abweichungen treten allerdings nur bei hohen Fehlmassenströmen auf. Die Einordnung eines Berechnungsergebnisses als kritisch (>10 t/h) würde an dem Beispieltage mit den neuronalen Netzen korrekt erfolgen. Dennoch sollte die Berechnung noch verbessert werden. Für die minimale Vorlauf-



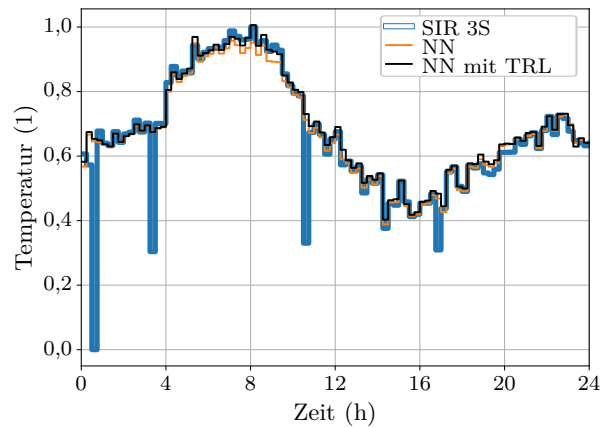
a) Massenstrom des HKW



b) Min. Δp in Teilgebiet (TG) 1 und 2



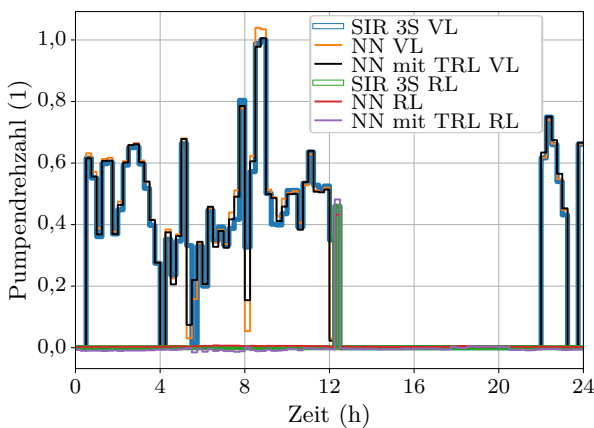
c) Fehlmassenstrom



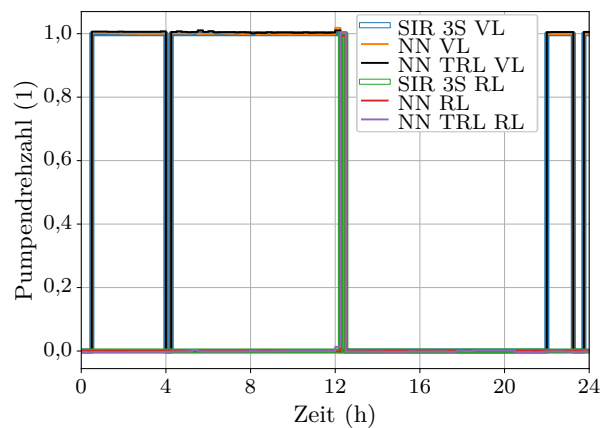
d) Min. Vorlauftemperatur in TG 1

Abbildung 6.20.: Vergleich normierter Simulationsergebnisse von Netz 3 an Tag 1 im Februar mit *SIR 3S*[®], dem neuronalen Netz (NN) ohne und mit Rücklauftemperatur (TRL).

temperatur sind in Abbildung 6.20a wieder die Ausreißer der Berechnungsergebnisse des *SIR 3S*[®] Modells zu sehen. Die Temperatur wird an den anderen Zeitpunkten gut berechnet.



a) Ohne Drehzahlvorgabe



b) Mit Drehzahlvorgabe

Abbildung 6.21.: Vergleich normierter Simulationsergebnisse für die Drehzahlen der DEA Vor- und Rücklaufpumpen (VL und RL) in Netz 3 an Tag 1 im Februar mit *SIR 3S*[®], dem neuronalen Netz (NN) ohne und mit Rücklauftemperatur (TRL).

Die Drehzahlen der DEA Vor- und Rücklaufpumpen sind in Abbildung 6.21 dargestellt. Abbildung 6.21a zeigt das Berechnungsergebnis ohne Drehzahlvorgabe. Für beide neuronalen Netze treten zu einigen Zeitpunkten Berechnungsfehler auf. In Abbildung 6.21b sind die Drehzahlen mit Vorgabe der minimalen Drehzahl dargestellt. In diesem Fall ist die Berechnung wie erwartet korrekt. In einer späteren Optimierung muss die Drehzahlvorgabe bei Drehzahlen kleiner als das Pumpenminimum aktiviert werden. Diese Einordnung funktioniert mit den neuronalen Netzen für alle Beispieltage. Die in Tabelle 6.26 markierten hohen Abweichungen der Pumpendrehzahlen der DEA sind also unkritisch.

Abbildung 6.22 zeigt, dass der minimale Differenzdruck auch mit Drehzahlvorgabe größtenteils gut vorhergesagt wird. Dies ist auch für die anderen Ausgangsgrößen der Fall. Allerdings treten zeitweise, auch in *SIR 3S*[®], fehlerhafte Ergebnisse auf. An Beispieltag 1 wird nicht immer eine Lösung der hydraulischen und thermischen Gleichungen gefunden, welche alle Restriktionen erfüllt. So können neuronale Netze diese Zeitpunkte nicht korrekt erlernen. Auch die Messdaten des Wintertages zeigen Betriebsartenwechsel der DEA, da Grenzwerte häufig erreicht wurden. Die Ergebnisse der neuronalen Netze sind also trotz Abweichungen nicht als schlecht zu beurteilen und können in einer Optimierung verwendet werden, um andere Fahrweisen der DEA und Wärmeerzeugungsanlagen zu finden.

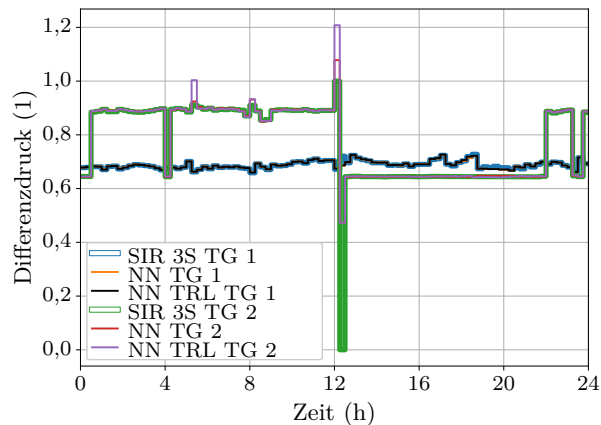
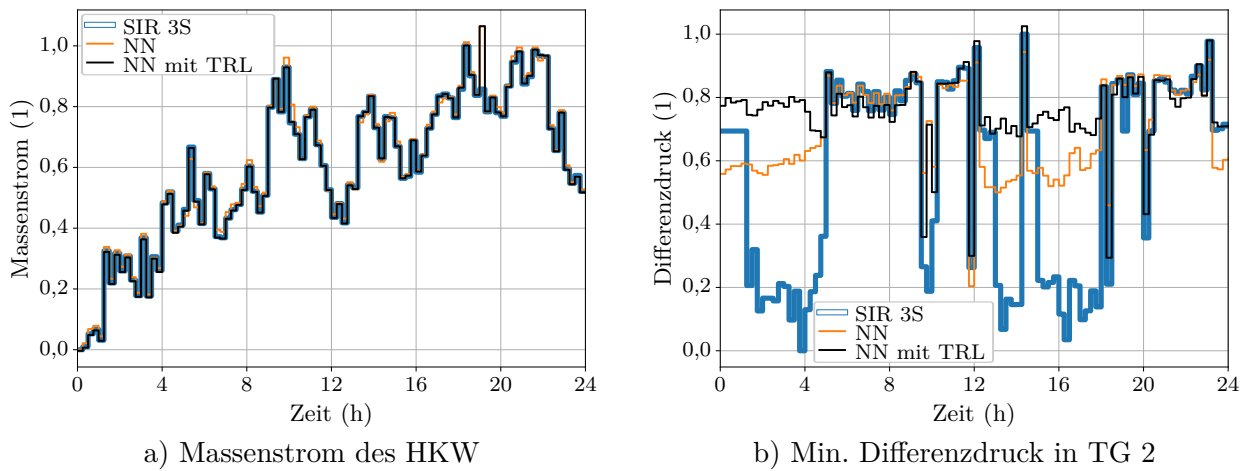


Abbildung 6.22.: Simulierte, normierte min. Differenzdrücke in TG 1 und TG 2 von Netz 3 an Tag 1 mit *SIR 3S*[®], dem neuronalen Netz (NN) ohne und mit Rücklaufftemperatur (TRL).

Für den Sommertag können die Ausgangsgrößen mit den neuronalen Netzen teilweise sehr gut nachgestellt werden, wie zum Beispiel der Graph des Massenstroms des HKW in Abbildung 6.23a zeigt. Lediglich zu einem Zeitpunkt um 19 Uhr treten größere Abweichungen auf, welche durch einen hohen Fehlmassenstrom und nicht plausible Simulationsergebnisse in *SIR 3S*[®] verursacht werden. Der Großteil der Ausgangsgrößen kann ähnlich gut vorausgesagt werden. Die relativen Abweichungen sind, wie bereits erwähnt, im Vergleich zum Wintertag höher, was in allgemein niedrigen absoluten Werten begründet ist. Dennoch können einige der Ausgangsgrößen, wie zum Beispiel der minimale Differenzdruck in Teilgebiet (TG) 2, dargestellt in Abbildung 6.23b, nicht gut berechnet werden. Aufgrund dieser fehlerhaften Berechnungen ist auch eine Optimierung für wärmere Tage problematisch, da Restriktionen teilweise nur aus diesem Grund nicht eingehalten werden und so keine Lösung gefunden wird.

Zusammenfassend können neuronale Netze Fernwärmenetz 3 für Tage mit hohem Wärmebedarf sehr gut abbilden und die Berechnungsergebnisse von *SIR 3S*[®] werden nahezu exakt getroffen. Für einige Größen wie zum Beispiel die Berechnung der DEA und des Fehlmassenstroms können die Netze bei Bedarf noch verbessert werden. Das neuronale Netz mit Rücklaufftemperaturerinput berechnet bessere Ergebnisse als das Netz ohne diese Information. Für Tage mit niedrigem Wärmebedarf nimmt die relative Genauigkeit der neuronalen Netze ab, während die absoluten Fehler nur leicht zunehmen. Auch diese Tage können also generell gut berechnet werden. Allerdings treten an dem Sommertag



a) Massenstrom des HKW

b) Min. Differenzdruck in TG 2

Abbildung 6.23.: Vergleich normierter Simulationsergebnisse von Netz 3 an Tag 3 im Juli mit *SIR 3S*[®], dem neuronalen Netz (NN) ohne und mit Rücklaufftemperatur (TRL).

zum Beispiel für die Berechnung des minimalen Differenzdrucks Fehler auf, welche problematisch für eine spätere Optimierung erscheinen. Hier könnten die neuronalen Netze durch Erzeugen weiterer Trainingsdaten mit niedrigen Wärmeverbräuchen nachgebessert werden.

6.4. Fernwärmenetz 4

Das letzte untersuchte Fernwärmenetz ist Netz 4, dargestellt in Abbildung 6.24, welches aus einem HKW inklusive Geothermieanlage und einem HW mit Wärme versorgt wird. Überschüssige Wärme wird in einer Speicheranlage gespeichert und bei zu geringer Einspeisung den Speichern wieder entnommen. An diesem Standort befinden sich auch zusätzliche Pumpen für die Druck- und Differenzdruckregelung von drei Teilgebieten. Zusätzlich sind zwei DEAs mit Pumpen im Vor- und Rücklauf vorhanden, welche für entfernte Gebiete den Druck erhöhen können. Weiterhin verbindet eine in der Abbildung nicht dargestellte Umformstation (UFS) ein weiteres Wohngebiet mit dem Fernwärmenetz.

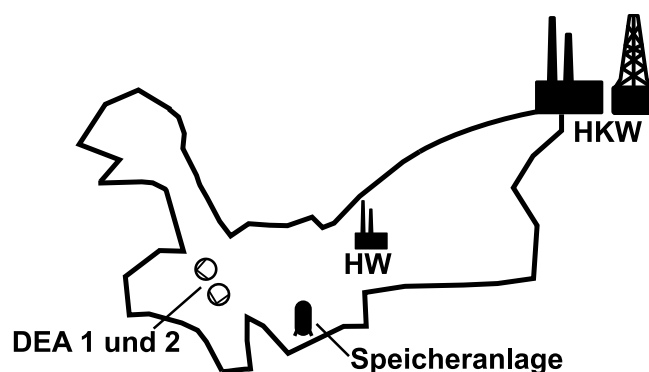


Abbildung 6.24.: Vereinfachte, nicht-maßstäbliche Darstellung des realen Fernwärmenetzes 4 mit Druckerhöhungs-, Speicher- und Wärmeerzeugungsanlagen.

Die in den vorherigen Kapiteln erarbeiteten Erkenntnisse zur Modellbildung und -auswertung werden auch auf Fernwärmenetz 4 angewandt. Da kein neues Vorgehen zum Einsatz kommt, sind die Informationen für das Modell in *SIR 3S*[®] und für neuronale Netze in Anhang A.1.5 vorhanden. Zusammenfassend sind in der folgenden Tabelle 6.27 die mittleren 24 Stunden Druck- und Temperaturab-

weichungen des *SIR 3S*[®] Modells zu den verfügbaren Messwerten dargestellt. Die Auswertung wurde für einen Wintertag im Februar (Tag 1) und einen Maitag (Tag 2) durchgeführt. Eine quasi-stationäre 24-Stunden-Simulation mit 15-Minuten-Zeitschritten dauert dabei einige Minuten.

Tabelle 6.27.: Mittlere 24h-Abweichungen der Nachberechnung von Fernwärmenetz 4 in *SIR 3S*[®] an den drei Beispieltagen.

	Tag 1	Tag 2
Druck (bar)	0,23	0,18
Standardabweichung Druck (bar)	0,20	0,14
Druck (%)	4,0	2,9
Temperatur (°C)	2,96	3,07
Standardabweichung Temperatur (°C)	3,79	2,96
Temperatur (%)	3,2	4,0

Des Weiteren zeigt Tabelle 6.28 die durchschnittlichen Abweichungen der zur Modellvereinfachung erstellten neuronalen Netze mit und ohne Rücklaufftemperatur verglichen zu *SIR 3S*[®] für Drücke und die entsprechend Tabelle A.22 relevanten Ausgangsgrößen für die Testdaten und zwei Beispieltage.

Tabelle 6.28.: Durchschnittliche Abweichung der neuronalen Netze zu *SIR 3S*[®] für Fernwärmenetz 4

	<i>MAPE</i> Drücke (%)	<i>MAE</i> Drücke (bar)	<i>MAPE</i> gesamt (%)
Neuronales Netz mit Rücklaufftemperatur			
Testdaten	0,067	0,004	0,167
Zehn Simulationstage basierend auf Tag 1	0,076	0,005	0,080
Zehn Simulationstage basierend auf Tag 2	0,135	0,009	0,523
Neuronales Netz ohne Rücklaufftemperatur			
Testdaten	0,084	0,004	0,181
Zehn Simulationstage basierend auf Tag 1	0,196	0,011	0,492
Zehn Simulationstage basierend auf Tag 2	0,124	0,008	0,961

Für den Großteil der Ausgangsgrößen der neuronalen Netze sind die Abweichungen sehr gering. Die Vorhersage des Speicherinhalts muss allerdings überarbeitet werden und funktioniert nicht ausreichend gut. Auch einige Pumpendrehzahlen und -leistungen können nicht gut vorhergesagt werden. Für die Pumpenleistungen ist dies nicht kritisch, da sie in den späteren Optimierungen nicht verwendet werden und nur bei weiterführenden Leistungsoptimierungen zum Einsatz kommen könnten. Auch die Abweichungen der Pumpendrehzahlen an Wintertagen sind nicht problematisch. Hier liegen die Drehzahlen generell in einem akzeptablen Bereich und alle restlichen Größen werden mit sehr guter Genauigkeit bestimmt. Für wärmere Tage sind die hauptsächlich durch Ausreißer entstehenden Abweichungen allerdings nicht akzeptabel. An dieser Stelle müssen die neuronalen Netze verbessert werden.

6.5. Diskussion der Modellierungsergebnisse

Abschließend werden hier die Ergebnisse und Limitierungen der Modelle diskutiert. Zuerst wurde ein einfaches Fernwärmenetz mit einer einzelnen Einspeisung als Basis für den Vergleich mehrerer Softwareoptionen verwendet. Stationäre und instationäre Simulationsmodelle wurden erstellt und die Firma 3S Consult GmbH für die weitere Modellierung ausgewählt. Physikalische Netzmodelle in *SIR 3S*[®] wurden anschließend für drei komplexere Fernwärmenetze gebildet. Anhand von Netz 2 konnte die dezentrale Wärmeeinspeisung mit einem simplen Beispiel mit nur zwei Wärmeeinspeisungen in ein großes Netz getestet werden. Netz 3 ist das größte in dieser Arbeit verwendete Fernwärmenetz, welches neben vier Wärmeeinspeisungen zwei Geothermieanlagen und eine komplexe Steuerung einer DEA aufweist. Für Netz 4 werden Modelle im Anhang vorgestellt, da keine wesentlichen Neuerungen hinzukommen.

Mit den von 3S Consult erstellten *SIR 3S*[®] Modellen konnte eine gute Übereinstimmung zu Messdaten an den betrachteten zwei bis vier Beispieltagen zu unterschiedlichen Jahreszeiten erreicht werden. Der Vergleich an Tagen mit hohem Wärmeverbrauch und entsprechend großen Ausgangswerten zeigt vor allem bezogen auf die hydraulischen Größen bessere Ergebnisse als der Vergleich an Tagen mit niedrigem Wärmeverbrauch. Eine mögliche Ursache sind durch die geringen Massenströme verursachte unklare Strömungsrichtungen in Maschen, welche auch zu langsamerer Konvergenz der Lösungsalgorithmen führen. Die Simulationsdauer für quasi-stationäre Tagessimulationen im 15-Minuten-Takt liegt so zwischen mehreren Sekunden für Tage mit hohem und Minuten für Tage mit niedrigem Wärmeverbrauch. Mit Fernwärmenetz 1 und 2 wurde untersucht, Simulationszeiten durch eine Aggregation der Rohrleitungen und Wärmeverbraucher zu erzielen. Für Netz 1 führt eine Aggregation auf etwa 20 % der ursprünglichen Knotenzahl zu erheblichen Abweichungen der Wärmeverluste und minimalen Differenzdrücke bei einem Vergleich zwischen aggregiertem und vollständigem Netzmodell. Netz 2 wurde daher nur noch auf etwa 60 % der Knoten aggregiert. Die resultierenden Simulationszeiten sind für den Einsatz in einer Optimierung oder MPC dennoch zu langsam. Hier bestätigt sich die bereits im Stand der Technik aus [88] zitierte Schwierigkeit, einen Kompromiss zwischen Genauigkeit und Schnelligkeit der Netzsimulation für eine MPC zu finden. Den physikalischen Netzmodellen sind klare Grenzen gesetzt.

Im Folgenden wurde die Vereinfachung stationärer Netzsimulationen durch Algorithmen des maschinellen Lernens untersucht. Für die weniger komplexen Fernwärmenetze 1 und 2 wurde zunächst ein Vergleich der Algorithmen lineare Regression, Polynomregression, *Random Forests*, XGBoost und neuronale Netze durchgeführt. In beiden Vergleichen liefern neuronale Netze den besten Kompromiss zwischen Genauigkeit und Rechenzeit bei vergleichsweise geringer Neigung zur Überanpassung an die mit *SIR 3S*[®] erstellten Trainingsdaten. Anschließend wurden die Parameter der neuronalen Netze in *PyTorch* untersucht, um eine möglichst genaue Approximation der *SIR 3S*[®] Simulationen zu erreichen. So wurden *Loss*- und Aktivierungsfunktionen sowie die Initialisierung der Gewichte, die Art der Lerndatennormierung und die Anzahl der versteckten Schichten sowie Neuronen variiert. Als Optimierer wurde *Adam* mit konstanter oder abnehmender Lernrate getestet. Auch der *Dropout* einiger Neuronen während des Trainings wurde für Fernwärmenetz 1 getestet, hat aber zu keiner signifikanten Verbesserung der Lernergebnisse geführt. Andere Optimierer oder Optionen wie beispielsweise die Regularisierung der Gewichte, das *Early-Stopping* des Trainings oder die Verwendung weiterer *Loss*-Funktionen wurden nicht untersucht. Mit diesen Ansätzen könnten die neuronalen Netze, falls notwendig, noch verbessert werden.

Die Erkenntnisse wurden im Anschluss erfolgreich auf das komplexe Fernwärmenetz 3 mit vier verteilten, teils regenerativen Wärmeerzeugungsanlagen angewandt. Insgesamt werden mit neuronalen Netzen für alle vier Fernwärmenetze durchschnittliche Abweichungen von unter 1 % zu *SIR 3S*[®] erreicht. Voraussetzung ist dabei die Vorgabe der Rücklauftemperaturen. Ohne diese erhöhen sich Abweichungen im Durchschnitt auf bis zu 2 % und lokale Ausreißer nehmen zu. Eine Prognose der Rücklauftemperatur wurde für Fernwärmenetz 2 erfolgreich mit neuronalen Netzen untersucht und in

das Prognosemodell der Netzzustände aufgenommen. Für die restlichen Fernwärmenetze wurde eine Rücklaufftemperaturprognose bisher nicht durchgeführt, ist prinzipiell aber mit dem gleichen Ansatz möglich. Für einzelne Ausgangsgrößen der neuronalen Netze entstehen größere relative und absolute Abweichungen. Diese treten vor allem an Sommertagen mit niedrigem Wärmeverbrauch auf und wurden in den jeweiligen Kapiteln beschrieben. Eine Verbesserung durch zusätzliche Trainingsdaten kann bei Bedarf untersucht werden. Die Rechenzeiten für Tagessimulationen mit 15-Minuten-Zeitschritten liegen für die vier Fernwärmenetze unter 1 ms.

Bei Einsatz der neuronalen Netze in einer Optimierung muss sichergestellt sein, dass diese stets korrekte Prognoseergebnisse unabhängig vom Betriebszustand der Fernwärmenetze liefern. Durch die Validierung der neuronalen Netze mit Testdaten und den Vergleich an den Beispieltagen konnte ein guter Status erreicht werden. Allerdings werden die Fernwärmenetze, wie bereits in Kapitel 4.2.2 festgelegt, nur vollständig modelliert. Das heißt, Ausnahmezustände wie die Absperrung einzelner Teilgebiete werden nicht berücksichtigt. Auch bauliche Änderungen an den Netzen, etwa hinzukommende Rohrleitungen oder Wärmeerzeugungsanlagen, können nicht direkt simuliert werden. Eine Implementierung dieser Zustände in die neuronalen Netze ist aber möglich. In *SIR 3S*[®] können so etwaige Ausnahmezustände und bauliche Änderungen in die Netzmodelle integriert und Simulationsdaten erzeugt werden. Diese können anschließend wieder von neuronalen Netzen erlernt werden. So können auch zukünftige Zustände der Fernwärmenetze mit zusätzlichen Versorgungsgebieten oder weiteren regenerativen Wärmeerzeugungsanlagen modelliert und simuliert werden. Eine Integration in eine spätere Optimierung und somit der Test von Ausbaustrategien ist ebenfalls möglich.

Weiterhin wichtig ist die Verteilung der Wärmeabnahme im Fernwärmenetz. In *SIR 3S*[®] wird bisher eine vorgegebene gesamte Wärmeabnahme durch fixe Annahmen der vorhandenen Verbraucher (z. B. Einfamilien- oder Mehrfamilienhäuser und Industriekunden) und deren Lastprofilen auf das Netz verteilt. Die Verteilung wurde dabei so gewählt, dass die *SIR 3S*[®] Modelle an den Beispieltagen eine möglichst gute Übereinstimmung zu den Messwerten liefern. Ausnahmen sind auch hier möglich. Für eine vollständige Validierung sowohl der *SIR 3S*[®] Netzmodelle als auch der neuronalen Netze ist eine kontinuierliche Überprüfung der Modelle innerhalb eines Jahreszyklus zu empfehlen. In dieser Zeit können notwendige Ausnahmezustände nachgearbeitet und die Modelle wo nötig verbessert werden.

Zudem können in weiteren Forschungsarbeiten Ansätze untersucht werden, die Modellierung durch neuronale Netze zu vereinfachen. So könnten zum Beispiel separate Modelle der einzelnen Teilgebiete eines Fernwärmenetzes erstellt und verknüpft werden. In diesem Fall wäre eine einfachere Berücksichtigung von Teilgebietsabsperrungen oder dem Hinzukommen neuer Leitungen und Wärmeerzeugungsanlagen möglich. Auch die Anzahl der Trainingsdaten könnte für eine schnellere Datenerzeugung in *SIR 3S*[®] reduziert werden. In Kapitel 6.1.3 konnte bereits gezeigt werden, dass das Training neuronaler Netze auch bei Verwendung einer geringeren Anzahl an Trainingsdaten nicht zu starker Überanpassung an diese führt. In einer weiterführenden Arbeit kann dieses Thema im Detail untersucht werden.

Des Weiteren wurde für Fernwärmenetz 1 ein Vergleich von stationären und instationären Simulationen in *SIR 3S*[®] durchgeführt. Bei geringer Veränderung der Vorlauftemperatur während eines Tagesverlaufs kommt es nur zu kleinen Abweichungen. Bei an Außentemperaturkurven angepassten Fahrweisen kann so zunächst davon ausgegangen werden, dass stationäre Simulationen ausreichen. Dies bestätigt die Validierung der vier stationären *SIR 3S*[®] Netzmodelle mit Messdaten an den Beispieltagen. Problematisch ist allerdings eine dynamische Änderung der Vorlauftemperaturen innerhalb kurzer Zeiträume wie in Kapitel 6.1.4 beschrieben. Eine solche Fahrweise könnte beispielsweise verwendet werden, um Wärmeverluste im Tagesverlauf zu minimieren. In diesem Fall zeigt der Vergleich zwischen stationären und instationären Simulationen deutliche Abweichungen und teils Verletzungen der Grenzwerte. Daher wurde der Ansatz der Systemidentifikation getestet, um instationäre Simulationen zu approximieren. Eine grundlegende Untersuchung zeigt, dass auch dies in geeigneten Modellen mit Rechenzeiten für einen Tagesverlauf mit 1-Minute-Zeitschritten im Bereich von 150 ms resultiert. Neuronale Netze wurden

ebenso für diesen Zweck getestet. Bisher wurden aber keine guten Ergebnisse erzielt. Beide Ansätze können daher in weiterführenden Forschungsvorhaben genauer untersucht werden.

7. Optimierung der Fernwärmenetze

Physikalische Knoten-Kanten-Modelle in der Software *SIR 3S*[®] für vier Fernwärmenetze stehen nun zur Verfügung. Die Modelle können thermisch stationär und instationär simuliert werden. Stationäre Modelle der vier Netze wurden durch neuronale Netze nachgebildet und zeigen im Vergleich mit *SIR 3S*[®] gute Berechnungsergebnisse. Für Fernwärmenetz 1 wurden zudem Systemidentifikationsmodelle für thermisch instationäre Simulationen erstellt.

Basierend auf den Modellen soll im Folgenden das Regelungs- bzw. Optimierungskonzept nach Abbildung 4.3 in Kapitel 4 umgesetzt werden. So wird eine Optimierung getestet, welche vorab nach wirtschaftlichen Kriterien optimierte Wärmeerzeugungsleistungen und Vorlauftemperaturen als Sollwerte erhält und diese möglichst direkt umsetzt. Bei Verletzung thermischer oder hydraulischer Restriktionen, welche durch die vorgestellten Simulationsmodelle berechnet werden, muss eine Umplanung der Leistungen und Temperaturen stattfinden. Dafür werden Optimierungen getestet, welche als Zielfunktion die Abweichungen der Leistungen und Vorlauftemperaturen zu deren wirtschaftlich optimalen Sollwerten erhalten, den vorgegebenen Wärmeverbrauch im System decken und die folgenden Nebenbedingungen (Restriktionen) einhalten:

- Minimale und maximale Drücke im Rohrleitungssystem
- Minimale Differenzdrücke der Verbraucher
- Maximale Strömungsgeschwindigkeiten
- Minimale Vorlauftemperaturen bzw. Einhaltung gegebener Vorlauftemperaturkurven und Gradienten
- Maximale Massenströme, Pumpendrehzahlen und Leistungen der Wärmeerzeugungsanlagen

Zunächst wird eine Optimierung untersucht, welche direkt die Netzmodelle in *SIR 3S*[®] verwendet. Dabei ist es nicht möglich, Gradienten der Zielfunktionen oder Restriktionen zu formulieren. Für eine Optimierung muss somit eine große Anzahl an Simulationen durchgeführt werden, welche ohnehin, verglichen mit neuronalen Netzen oder Modellen der Systemidentifikation viel Zeit in Anspruch nehmen. Dieser Ansatz ist daher nicht zielführend.

Stattdessen werden die Untersuchungen im Folgenden mit den vereinfachten Modellen durchgeführt. Netz 1 wird wie für die Modellierung als einfaches Beispiel verwendet. Da nur ein Wärmeerzeugungsstandort vorhanden ist, ist eine Optimierung der verteilten Wärmeeinspeisung nicht möglich. Allerdings sind an dem Standort vier Wärmeeinspeisungen inklusive zwei Speicheranlagen installiert, zwischen welchen die Leistungs- und Vorlauftemperaturverteilung bezüglich wirtschaftlicher Sollwerte optimiert werden kann. Diese Optimierung wird anhand eines vereinfachten, in Kapitel 7.1.1 vorgestellten, thermo-hydraulischen Standortmodells getestet. Die Ergebnisse der Gesamtleistung und Vorlauftemperatur werden anschließend auf das Netzmodell angewandt, um zu verifizieren, dass auch dort alle thermo-hydraulischen Restriktionen eingehalten werden. Als Netzmodell wird das in Kapitel 6.1.3 erstellte neuronale Netz verwendet. Zudem kann die insgesamt ins Netz eingespeiste Vorlauftemperatur, abweichend von wirtschaftlichen Sollwerten, verändert werden. Hier wird eine Minimierung der Wärmeverluste durch eine Absenkung der Vorlauftemperatur angestrebt. Für die Verifikation der

Netz-Restriktionen wird zunächst wieder das neuronale Netz als stationäres Simulationsmodell verwendet. In Kapitel 6.1.4 wurde bereits festgestellt, dass eine deutliche Veränderung der Vorlauftemperatur mit stationären Simulationen und Optimierungen zu größeren Fehlern führt. Daher wird für Netz 1 eine instationäre Optimierung der Temperatur unter Verwendung von Systemidentifikationsmodellen getestet. Insgesamt werden so mit Fernwärmenetz 1 als Beispiel erste Optimierungen mit neuronalen Netzen und Systemidentifikationsmodellen erarbeitet.

Fernwärmenetz 2 mit zwei Wärmeerzeugungsanlagen wird in Kapitel 7.2 als einfacher Testfall für die Optimierung der verteilten Wärmeerzeugung verwendet. Eine Optimierung mit gradientenbasierten Ansätzen und neuronalen Netzen als Simulationsmodell wird in Kapitel 7.2.1 beschrieben und ausgewertet. In Kapitel 7.2.2 wird zudem untersucht, ob die Optimierung bei abweichenden Wärmeabnahmegprognosen zuverlässig Ergebnisse liefert. Sämtliche Berechnungen werden mit neuronalen Netzen durchgeführt und instationäre Effekte daher nicht simuliert. Kapitel 7.2.3 beschreibt das Vorgehen, um die Vorlaufleitungen auch ohne Berücksichtigung instationärer Simulationen als Temperaturspeicher zu verwenden. Zuletzt wird eine weitergehende Optimierung mit dem Ziel der Minimierung der Wärmeverluste grundlegend in Kapitel 7.2.4 beschrieben.

In Kapitel 7.3 wird das vorgestellte Optimierungskonzept an dem komplexen Fernwärmenetz 3 mit vier verteilten Wärmeerzeugungsanlagen getestet. Durch eine Geothermieanlage in einem Teilnetz entstehen zusätzliche Restriktionen. Das Teilnetz ist zudem durch eine DEA an das Fernwärmenetz angeschlossen, welche in Kapitel 6.3 mit fünf Betriebsarten und der Möglichkeit der Drehzahlvorgabe modelliert wurde. Die Betriebsart ist vor einer Optimierung nicht bekannt. Die direkte Berücksichtigung inklusive nicht-linearer thermo-hydraulischer Restriktionen führt wiederum zu einer schwer lösbaren gemischt-ganzzahligen nicht-linearen Optimierung. Die Fahrweise der DEA wird daher zunächst in Kapitel 7.3.1 analysiert. Anschließend wird in Kapitel 7.3.2 vorgestellt, wie die Betriebsarten in die Optimierung einbezogen werden und eine möglichst gute Einhaltung wirtschaftlicher Sollwerte möglich ist. Alle Optimierungen werden mit einem neuronalen Netz als stationärem Simulationsmodell durchgeführt. Aufgrund der durch mehrere verteilte Wärmeeinspeisungen zunehmenden Anzahl an Variablen werden für Fernwärmenetz 3 globale Optimierungsalgorithmen getestet. In Kapitel 7.3.3 wird wie für Netz 3 das Verhalten der Optimierung bei abweichender Wärmeabnahmegprognose untersucht. Auch die Themen der Vorlauftemperaturspeicherung sowie der Wärmeverlustminimierung werden erneut in den Kapiteln 7.3.4 und 7.3.5 behandelt. Die Optimierung von Fernwärmenetz 4 verwendet das gleiche Vorgehen wie Fernwärmenetz 3, resultiert jedoch in keinen neuen Erkenntnissen und wird daher im Anhang A.2 vorgestellt.

Alle folgenden Untersuchungen werden auf einem Rechner mit *Intel Core i7-11800H* (8 x max. 4,60 GHz) und 16 GB RAM durchgeführt. Die Simulationen durch neuronale Netze und Systemidentifikationsmodelle sowie Optimierungen werden dabei ausschließlich durch die *CPU* durchgeführt. Eine Verwendung einer *Nvidia GeForce RTX 3060* für die neuronalen Netze führt bisher nicht zu einer Beschleunigung der Optimierung. Hier konnte noch kein effizienter Austausch zwischen von der *CPU* gesteuerten Optimierungsalgorithmen und von der Grafikkarte berechneten neuronalen Netze gefunden werden.

7.1. Optimierung von Fernwärmenetz 1

Fernwärmenetz 1 hat, wie in Kapitel 6.1 beschrieben, nur einen Standort, an welchem zwei Wärmeerzeugungsanlagen und zwei Speicher zusammengeführt werden. Die gesamte Leistung der vier Anlagen mit deren Einzeltemperaturen ergibt die Vorlauftemperatur des Standorts, welche durch eine Beimischung aus dem Rücklauf abgekühlt werden kann. Aus der wirtschaftlichen Optimierung ergeben sich Sollwerte für die Leistungen der vier Anlagen und die Vorlauftemperatur wird entsprechend einer außen temperaturabhängigen Kurve gefahren. Da nur an einem Standort Wärme ins Netz gespeist wird

kann hier keine Optimierung mit im Netz verteilter Einspeisung untersucht werden. Allerdings können erste Optimierungstests mit den in Kapitel 6.1 vorgestellten neuronalen Netzen und Systemidentifikationsmodellen durchgeführt werden. Zudem wird die Wärmeeinspeisung an dem Standort selbst zwischen den vier Anlagen untersucht.

So kann nach der wirtschaftlichen Optimierung eine weiterführende physikalische Optimierung die Leistungen entsprechend der wirtschaftlichen Vorgabe und der gegebenen Vorlauftemperatur auf die einzelnen Anlagen verteilen. Hier soll durch eine vereinfachte Modellierung des Wärmeerzeugungsstandorts geprüft werden, ob mit den Sollleistungen, den einzelnen Vorlauftemperaturen und der Beimischung der Sollwert für die gesamte Vorlauftemperatur erreicht werden kann. Zusätzlich können Unterschiede zwischen der Prognose der benötigten Wärmeleistung und den tatsächlichen Messwerten entstehen. In diesem Fall passen die Sollwerte der wirtschaftlichen Optimierung nicht mehr zu der insgesamt notwendigen Wärmeleistung und die Leistungen müssen neu auf die vier Wärmeerzeugungsanlagen verteilt werden.

Weiterhin kann eine Vorlauftemperaturoptimierung, zum Beispiel zur Reduktion der Wärmeverluste, durchgeführt werden, um eine außentemperaturabhängige Kurve zu ersetzen. Bei dieser Optimierung muss darauf geachtet werden, dass alle hydraulischen und thermischen Restriktionen im Fernwärmenetz eingehalten werden. Ebenfalls verringert sich durch einen geringeren Wärmeverlust die insgesamt ins Netz einzuspeisende Leistung. Die Leistung würde also wieder von der Vorhersage abweichen und müsste neu auf die Wärmeerzeugungsanlagen verteilt werden. Im Optimierungsablauf würde die Vorlauftemperaturoptimierung somit vor der Leistungsoptimierung stehen. Insgesamt wäre der Optimierungsablauf der folgende:

1. Vorlauftemperaturoptimierung mit Einhaltung aller hydraulischen und thermischen Restriktionen im Netz: Als Resultat ergibt sich eine reduzierte Gesamtleistung.
2. Leistungsoptimierung: Verteilung der Gesamtleistung mit optimierter Vorlauftemperatur auf die einzelnen Wärmeerzeugungsanlagen am Standort, möglichst unter Einhaltung der wirtschaftlich optimalen Leistungen.

Im Folgenden werden zunächst die Leistungsoptimierung in Kapitel 7.1.1 und anschließend die Vorlauftemperaturoptimierung vorgestellt. Dafür kommt ein stationärer Ansatz in Kapitel 7.1.2 oder ein instationärer Ansatz in Kapitel 7.1.3 infrage.

7.1.1. Netz 1: Optimierung der Einspeiseleistungen

Bei der Leistungsoptimierung gilt es, die von der wirtschaftlichen Optimierung vorgegebenen Leistungen möglichst genau unter Einhaltung einer bestimmten Vorlauftemperatur auf die vier Wärmeerzeugungsanlagen in Fernwärmenetz 1 zu verteilen. Dies ist beispielsweise notwendig, wenn die prognostizierten Leistungen von der tatsächlichen Gesamtleistung im Netz abweichen und die Vorlauftemperatur passend eingestellt werden muss. Abbildung 7.1 zeigt für Beispieltag 1 (Januar) die vorhergesagte Gesamtleistung und die Gesamtleistung in *SIR 3S*[®]. *SIR 3S*[®] wurde an diesem Tag mit den gemessenen Prozessdaten validiert. Die Gesamtleistung der Simulation in *SIR 3S*[®] stimmt also gut mit der tatsächlich gefahrenen Leistung überein und weicht deutlich von der Prognose ab. Diese stammt aus einer 24-Stunden-Vorhersage vom Vortag. Mittlerweile steht auch eine Onlineversion der Vorhersage zur Verfügung, welche alle 15 Minuten aktualisierte Werte liefert. Die Abweichung sollte im tatsächlichen Betrieb also kleiner sein. Eine Umplanung kann aber dennoch bei kurzfristig abweichenden Leistungen notwendig sein.

Die Leistungsoptimierung wird anhand eines vereinfachten Stationsmodells mit physikalischen Gleichungen durchgeführt. So wird im Folgenden erläutert, wie ein einzelner Standort mit mehreren Wärmeerzeugungsanlagen auch mit einem einfachen mathematischen Modell optimiert werden kann. Ein

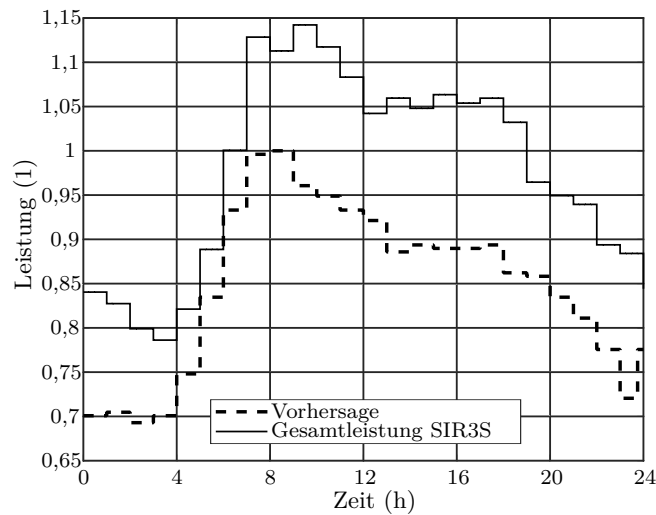


Abbildung 7.1.: In *SIR 3S*[®] simulierte, normierte Gesamtleistung in Fernwärmenetz 1 im Vergleich zur Vorhersage an Beispieltag 1 im Januar.

spezifisches Simulationsmodell, z. B. in *SIR 3S*[®], ist nicht notwendig. Die aufgestellten Gleichungen werden aber teilweise mit der erweiterten Stationsmodellierung des HKW in *SIR 3S*[®] verifiziert. Abbildung 7.2 zeigt eine vereinfachte Darstellung des HKW mit Zuleitung aus HKW 2 und zwei Speichern. Zusätzlich ist die Rücklaufbeimischung eingezeichnet.

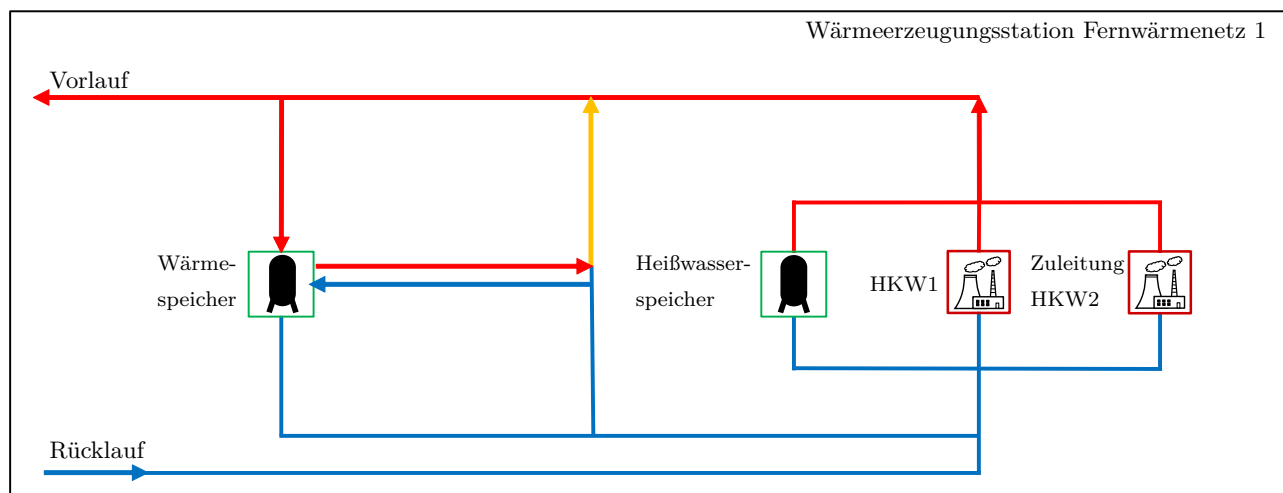
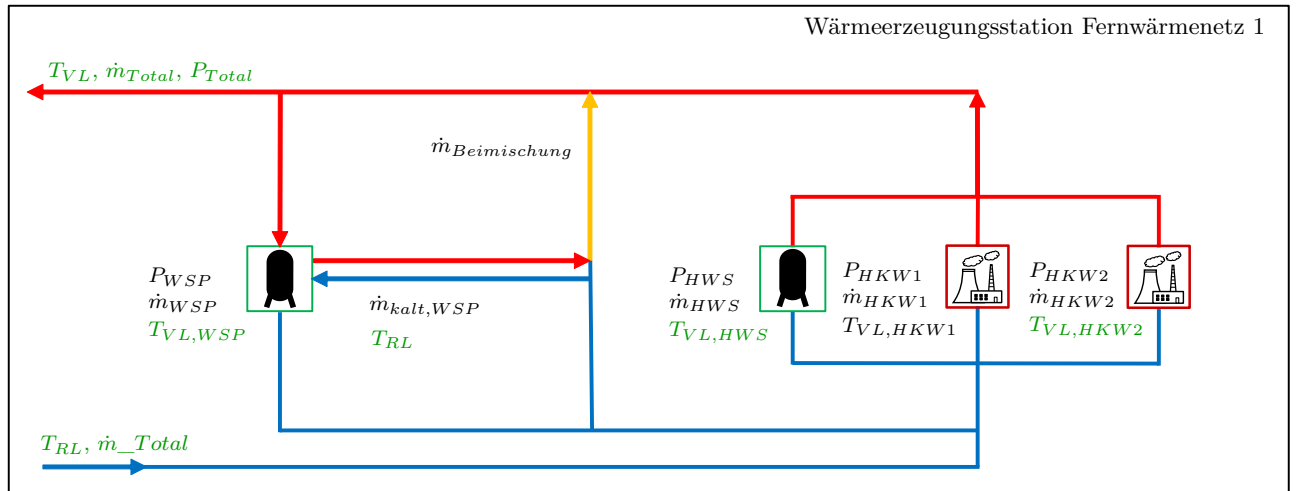


Abbildung 7.2.: Vereinfachte Darstellung der Wärmeerzeugungsstation in Fernwärmenetz 1.

In der vereinfachten Leistungsoptimierung wird zunächst eine *SIR 3S*[®] Simulation mit simpler Stationsmodellierung (aggregiert, quasi-stationär) und gegebener Vorlauftemperatur durchgeführt. Nach dieser ersten Simulation ist das Verhalten des Netzes (Massenströme, Drücke, Temperaturen) bei einer bestimmten Abnahmeleistung und Vorlauftemperatur vollständig bekannt. Nur noch die Leistungen und Temperaturen innerhalb der Wärmeerzeugungsstation müssen korrekt verteilt werden. In Abbildung 7.3 sind die nach der ersten Simulation bekannten Größen in Grün dargestellt. Die Vorlauftemperatur T_{VL} , die gesamte Leistung P_{Total} , der gesamte Massenstrom \dot{m}_{Total} und die Rücklauftemperatur T_{RL} ergeben sich aus der Simulation. Die Vorlauftemperaturen des Wärmespeichers und des Heißwasserspeichers werden vereinfacht konstant angenommen. Der Zulauf aus dem HKW 2 speist im Winter und Sommer mit zwei unterschiedlichen konstanten und dementsprechend bekannten Vorlauftemperaturen ein. Die anderen Größen müssen durch eine Optimierung berechnet werden.

Abbildung 7.3.: Wärmereizungsstation nach der ersten vereinfachten *SIR 3S*[®] Simulation.

Die Optimierung muss nun die Leistung P_{WSP} und den Massenstrom \dot{m}_{WSP} des Wärmespeichers bestimmen. Bei der Entladung fließt das Wasser dabei in den Massenstrom der Beimischung \dot{m}_{Bei} . Beim Beladen wird dem Wärmespeicher eine Mischung aus Wasser mit Vorlauf- und Rücklauf-temperatur zugeführt, um eine bestimmte bekannte Temperatur $T_{VL,WSP}$ zu erhalten. Des Weiteren sollen die Leistungen und Massenströme der anderen drei Anlagen (HKW 1, HKW 2 und Heißwasser- speicher) sowie die Temperatur des HKW 1 bestimmt werden. Die Optimierung erhält die Leistungen der vier Erzeuger (P_{WSP} , P_{HWS} , P_{HKW1} , P_{HKW2}), den Beimischungsstrom (\dot{m}_{Bei}) und die Vor- lauf-temperatur $T_{VL,HKW1}$ als Variablen. Diese können also verändert werden. Die physikalischen Zusammenhänge zwischen den Variablen und den weiteren Größen werden dem Optimierer als Restriktion übergeben. Als zu minimierende Zielfunktion wird die Abweichung der Leistungsvariablen (P) und der Sollwerte (SW) verwendet:

$$\Delta WSP = 100 \cdot (P_{WSP} - SW_{P_{WSP}}) \quad (7.1)$$

$$\Delta HWS = 100 \cdot (P_{HWS} - SW_{P_{HWS}}) \quad (7.2)$$

$$\Delta HKW1 = P_{HKW1} - SW_{P_{HKW1}} \quad (7.3)$$

$$\Delta HKW2 = P_{HKW2} - SW_{P_{HKW2}} \quad (7.4)$$

$$\Delta \mathbf{P} = [\Delta WSP; \Delta HWS; \Delta HKW1; \Delta HKW2] \quad (7.5)$$

$$f = 0,5 \cdot (\Delta \mathbf{P}' \cdot \Delta \mathbf{P}) \quad (7.6)$$

Die einzelnen Abweichungen ΔWSP , ΔHWS , $\Delta HKW1$ und $\Delta HKW2$ werden über eine Vektor- multiplikation zu dem Zielfunktionswert f zusammengefasst. Die Abweichungen der beiden Speicher ΔWSP und ΔHWS werden stärker gewichtet, um deren Leistungen genauer einzuhalten. Die Ge- wichtung kann angepasst werden, um andere Optimierungsziele zu erreichen. Bei der Optimierung müssen zwei Restriktionen eingehalten werden. Der Gesamtmassenstrom der Wärmereizungs- anlagen muss dem Gesamtmassenstrom aus der ersten *SIR 3S*[®] Simulation entsprechen. Zusätzlich muss die Mischtemperatur T_{Misch} , die sich aus den Temperaturen und Leistungen der Wärmereizungs- anlagen ergibt, gleich der vorher berechneten und vorgegebenen Vorlauf-temperatur T_{VL} sein:

$$\dot{m}_{Total} \stackrel{!}{=} \dot{m}_{Bei} + \dot{m}_{HWS} + \dot{m}_{HKW1} + \dot{m}_{HKW2} \quad (7.7)$$

$$T_{Misch} \stackrel{!}{=} T_{VL} \quad (7.8)$$

Für die Zusammenhänge in der Wärmeerzeugungsstation ergeben sich, je nachdem, ob der Wärmespeicher und der Heißwasserspeicher entladen oder beladen werden, andere physikalische Zusammenhänge. Die einzelnen Formeln, welche der Optimierer verwendet, sollen hier aufgeführt werden. Zuerst wird dabei auf die Gruppe der drei Wärmeerzeugungsanlagen mit dem Heißwasserspeicher eingegangen. Die Flusswege beim Entladen und Beladen sind in Abbildung 7.4 dargestellt.

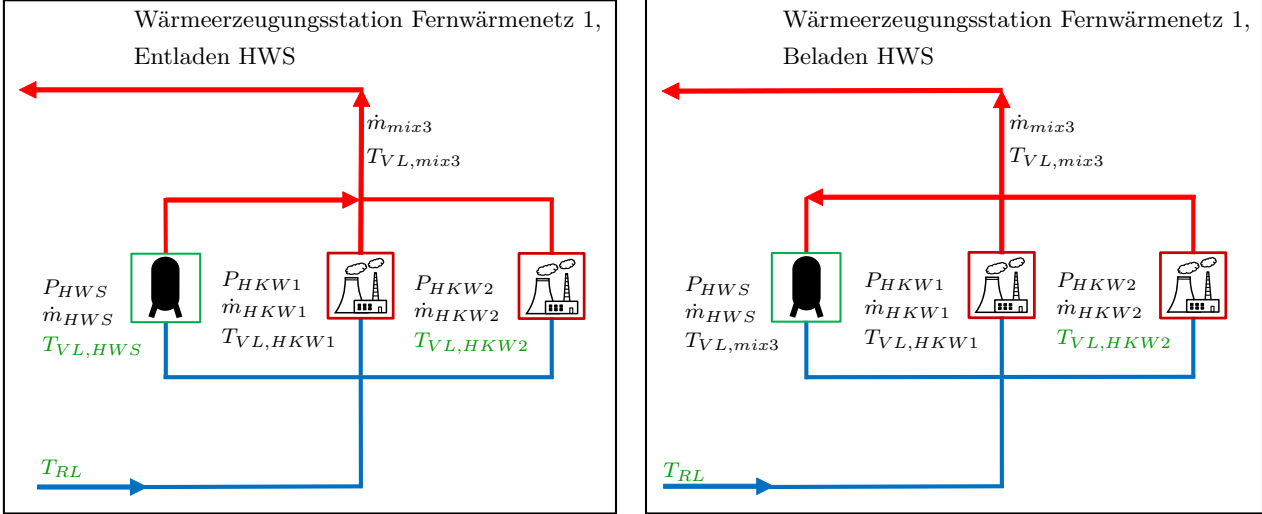


Abbildung 7.4.: Ent- und Beladen des Heißwasserspeichers in Fernwärmenetz 1.

Für beide Fälle gilt für die Massenströme aus dem HKW 2 und HKW 1:

$$\dot{m}_{HKW2} = \frac{P_{HKW2}}{c_p \cdot (T_{VL,HKW2} - T_{RL})} \quad (7.9)$$

$$\dot{m}_{HKW1} = \frac{P_{HKW1}}{c_p \cdot (T_{VL,HKW1} - T_{RL})} \quad (7.10)$$

Beim Entladen des Heißwasserspeichers wird dessen Massenstrom mit den anderen beiden Massenströmen gemischt. Als Hilfsgröße wird der Massenstrom \dot{m}_{mix3} mit der Temperatur $T_{VL,mix3}$ eingeführt:

$$\dot{m}_{HWS} = \frac{P_{HWS}}{c_p \cdot (T_{VL,HWS} - T_{RL})} \quad (7.11)$$

$$\dot{m}_{mix3} = \dot{m}_{HWS} + \dot{m}_{HKW1} + \dot{m}_{HKW2} \quad (7.12)$$

$$T_{VL,mix3} = \frac{(\dot{m}_{HWS} \cdot T_{VL,HWS}) + (\dot{m}_{HKW1} \cdot T_{VL,HKW1}) + (\dot{m}_{HKW2} \cdot T_{VL,HKW2})}{\dot{m}_{mix3}} \quad (7.13)$$

Beim Beladen des Heißwasserspeichers wird dieser mit der Mischtemperatur von HKW 1 und HKW 2 befüllt. Die Leistung P_{HWS} und der Massenstrom \dot{m}_{HWS} sind in diesem Fall negativ. Die Mischtemperatur und der Massenstrom ergeben sich wie folgt:

$$T_{VL,mix3} = \frac{(\dot{m}_{HKW1} \cdot T_{VL,HKW1}) + (\dot{m}_{HKW2} \cdot T_{VL,HKW2})}{\dot{m}_{HKW1} + \dot{m}_{HKW2}} \quad (7.14)$$

$$\dot{m}_{HWS} = \frac{P_{HWS}}{c_p \cdot (T_{VL,mix3} - T_{RL})} \quad (7.15)$$

$$\dot{m}_{mix3} = \dot{m}_{HWS} + \dot{m}_{HKW1} + \dot{m}_{HKW2} \quad (7.16)$$

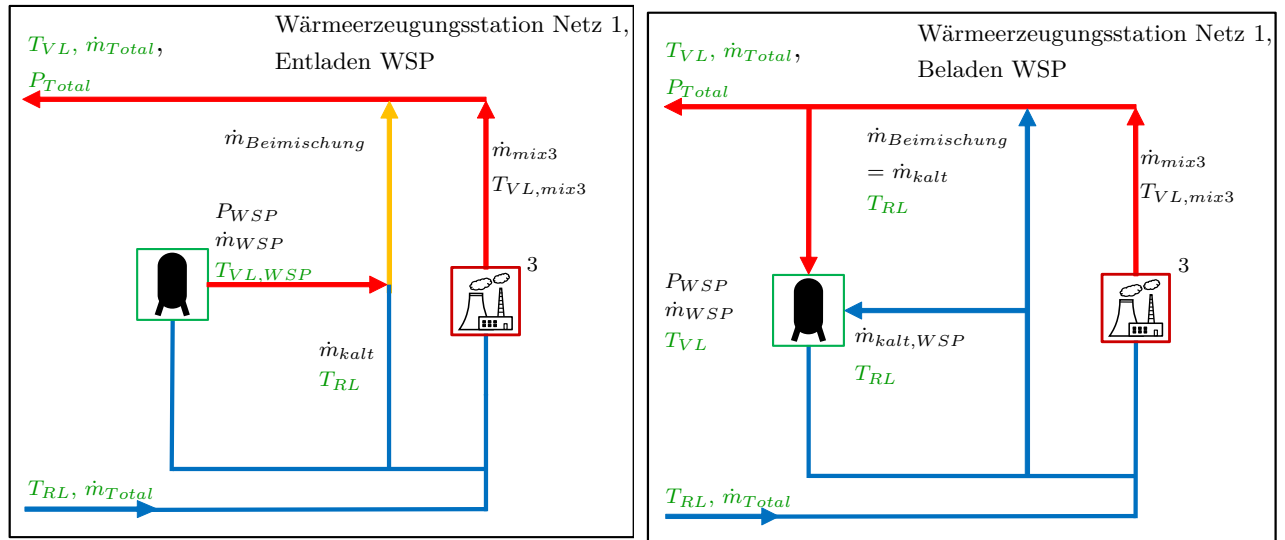


Abbildung 7.5.: Ent- und Beladen des Wärmespeichers in Fernwärmenetz 1.

Abbildung 7.5 zeigt das Ent- und Beladen des Wärmespeichers und die entsprechenden Flusswege. Die folgenden Formeln werden von der Optimierung für das Entladen des Wärmespeichers verwendet:

$$\dot{m}_{WSP} = \frac{P_{WSP}}{c_p \cdot (T_{VL,WSP} - T_{RL})} \quad (7.17)$$

$$T_{VL} = \frac{(\dot{m}_{WSP} \cdot T_{VL,WSP}) + (\dot{m}_{mix3} \cdot T_{mix3}) + (\dot{m}_{kalt} \cdot T_{RL})}{\dot{m}_{WSP} + \dot{m}_{mix3} + \dot{m}_{kalt}} \quad (7.18)$$

$$\dot{m}_{Bei} = \dot{m}_{kalt} + \dot{m}_{WSP} \quad (7.19)$$

Das Beladen, welches bei dem Wärmespeicher mit niedrigeren Temperaturen unter Beimischung eines kalten Massenstroms aus dem Rücklauf durchgeführt wird, lässt sich wie folgt berechnen:

$$T_{VL} = \frac{(\dot{m}_{mix3} \cdot T_{mix3}) + (\dot{m}_{kalt} \cdot T_{RL})}{\dot{m}_{mix3} + \dot{m}_{kalt}} \quad (7.20)$$

$$\dot{m}_{WSP} = \frac{P_{WSP}}{c_p \cdot (T_{VL} - T_{RL})} \quad (7.21)$$

$$\dot{m}_{Bei} = \dot{m}_{kalt} \quad (7.22)$$

Abbildung 7.6 zeigt das Ergebnis der Leistungsoptimierung für Beispieltag 1 im Januar. Die Optimierung wird in *MATLAB*[®] mit dem Solver WORHP oder in *Python* mit dem Optimierer *minimize* der Bibliothek *scipy* durchgeführt. Die Ergebnisse der Leistungsoptimierung sind mit „optimiert“ bezeichnet, die Sollwerte mit „Sollwert“. Für die Optimierung werden die vorhergesagten optimalen Leistungen und die tatsächlich gemessenen Verbrauchsdaten verwendet. Die vorhergesagten Verbräuche weichen dabei von den gemessenen Verbräuchen ab, wie bereits in Abbildung 7.1 gezeigt wurde.

Die hier vorgestellte Optimierung ist insofern kostenoptimiert, als dass die von der wirtschaftlichen Optimierung vorgegebenen Leistungen möglichst genau eingehalten werden. Bei Abweichungen zwischen vorhergesagter und tatsächlicher Leistung kann diese Abweichung nach bestimmten Gewichtungsfaktoren ausgeglichen werden. In der vorliegenden Optimierung werden die Leistungen der beiden Speicher stärker gewichtet und deshalb genau eingehalten, wie in Abbildung 7.6a ersichtlich ist. Das HKW 2 speist fast den gesamten Tag mit der maximalen Leistung ein, weshalb die fehlende Leistung zwischen Vorhersage und tatsächlicher Gesamtleistung nahezu vollständig von HKW 1 übernommen wird, welches eigentlich keine Leistung einspeisen soll. Die Gewichtungsfaktoren können anders gewählt werden, um zum Beispiel den Einsatz von HKW 1 zu minimieren. Abbildung 7.6b zeigt zudem

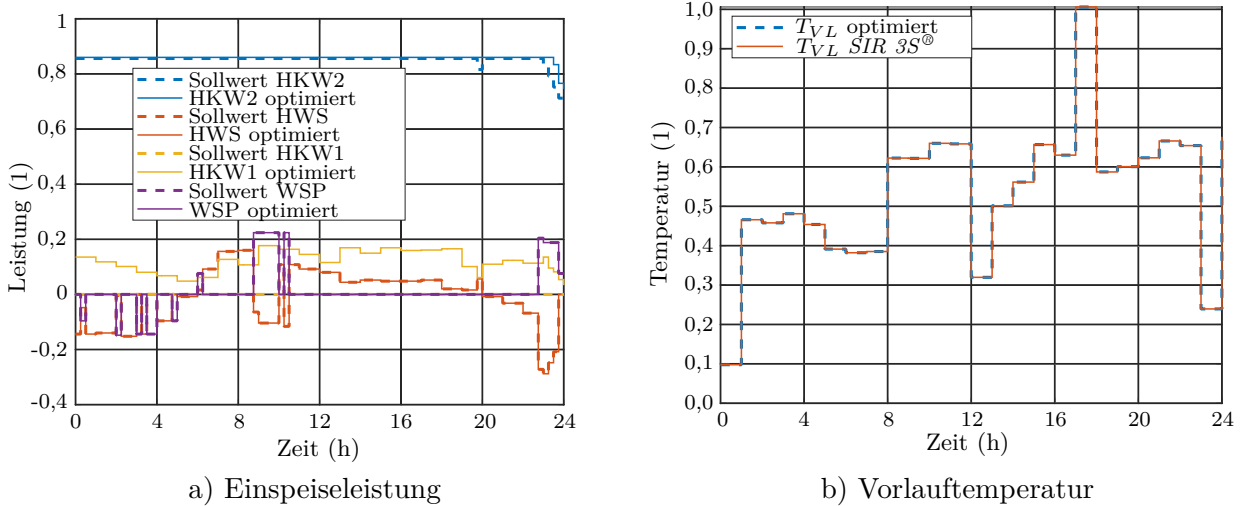
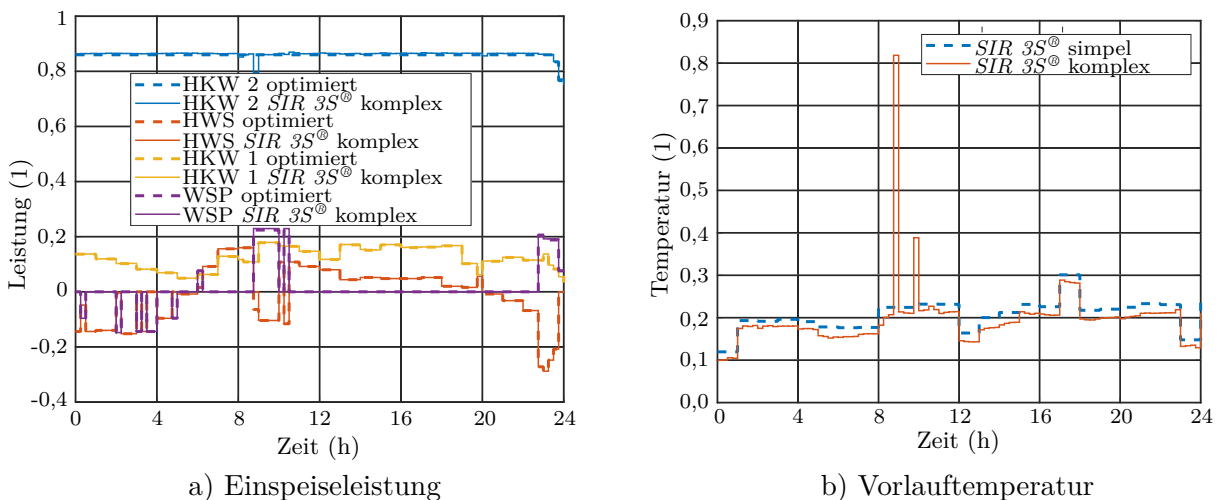


Abbildung 7.6.: Normierte Ergebnisse der Leistungsoptimierung in Fernwärmenetz 1.

die korrekte Einhaltung der Vorlauftemperatur nach der Optimierung. Der Sollwert und die optimierte Temperaturkurve liegen übereinander.

Die Optimierungsergebnisse werden anschließend mit der erweiterten Stationsmodellierung im *SIR 3S*[®] Modell verifiziert. Die Netzberechnung ist wie zuvor aggregiert und quasi-stationär eingestellt. Abbildung 7.7a zeigt jeweils die Leistungen aus der Optimierung („optimiert“) und die Ergebnisse, wenn diese Leistungen auf das *SIR 3S*[®] Modell mit erweiterter Stationsmodellierung angewandt werden („*SIR 3S*[®] komplex“). Zu fast allen Zeitpunkten werden die Leistungen sehr gut eingehalten. Lediglich um 8:45 Uhr kann die Leistung des Heißwasserspeichers nicht korrekt gefahren werden. Der Speicher wird zu diesem Zeitpunkt nicht wie vorgegeben beladen, weshalb das HKW 2 weniger Leistung einspeist. Abbildung 7.7b zeigt das Resultat der Vorlauftemperatur. Die Vorlauftemperatur aus der Leistungsoptimierung und das Ergebnis aus dem *SIR 3S*[®] Modell stimmen ebenfalls gut überein. Eine größere Abweichung ergibt sich zu dem Zeitpunkt, an welchem der Heißwasserspeicher in der erweiterten Stationsmodellierung nicht korrekt gefahren werden kann. Die Ergebnisse der Optimierung mit vereinfachten physikalischen Gleichungen können also größtenteils mit der erweiterten Stationsmodellierung in *SIR 3S*[®] validiert werden. Auftretene Abweichungen haben ihren Ursprung in Konvergenzproblemen der erweiterten Stationsmodellierung. Die vereinfachten physikalischen Formeln bilden die Wärmeerzeugungsanlagen hinreichend genau ab.

Abbildung 7.7.: Normierte Ergebnisse einer Validierungsrechnung mit erweiterter *SIR 3S*[®] Stationsmodellierung.

7.1.2. Netz 1: Stationäre Optimierung

In diesem Kapitel wird eine stationäre Vorlauftemperaturoptimierung mit dem Ziel, den Wärmeverlust im Netz unter Einhaltung von Restriktionen zu minimieren, vorgestellt. So kann eine erste Optimierung unter Verwendung von neuronalen Netzen als Simulationsmodell untersucht werden. Eine Minimierung des Wärmeverlusts ist durch ein Reduzieren der Vorlauftemperatur erreichbar. Dies führt zu einer geringeren Temperaturdifferenz zwischen Wasser, Rohrleitungen und Umgebung, jedoch zu höheren Strömungsgeschwindigkeiten. Bei kürzerer Verweildauer im Rohr kühlt das Wasser ebenfalls weniger ab. Mit dem in Kapitel 6.1.3 vorgestellten neuronalen Netz (NN) für quasi-stationäre Simulationen des vollständigen *SIR 3S*[®] Modells soll nun eine Vorlauftemperaturoptimierung für den Beispieltag 1 im Januar durchgeführt werden. In den quasi-stationären Simulationen werden dabei keine Netzspeichereffekte berücksichtigt. Die Zielfunktion der Optimierung ist ein möglichst niedriger Wärmeverlust. Dabei müssen die folgenden Restriktionen eingehalten werden:

- Nenndrehzahl als maximale Pumpendrehzahl (n_{Pumpe}) für die Umwälzpumpen in der Station
- Maximale Strömungsgeschwindigkeit (v) entlang der Haupttransporttrassen
- Minimaler Differenzdruck (Δp) an allen Kundenanlagen
- Minimale Vorlauftemperatur (T_{VL}) an allen Kundenanlagen
- Netzweiter minimaler Rücklaufdruck (p_{RL}) und maximaler Vorlaufdruck (p_{VL})
- Maximaler Netzeinspeisetemperaturgradient am Ausgang der Station

Der Dampfdruck wird nicht überprüft, da die Drücke in Fernwärmenetz 1 deutlich über diesem liegen. Die verwendeten Restriktionen basieren in Teilen auf Angaben des Netzbetreibers. Da aber nicht zu allen Grenzwerten Angaben vorliegen, werden sie mit dem Ziel, einen möglichst vollständigen und realistischen Rahmen zu gewährleisten, um eigene Annahmen ergänzt. Selbstverständlich bieten die Modelle (NN bzw. *SIR 3S*[®]) die Möglichkeit, die genannten Restriktionen umzuformulieren oder weitere Grenzwerte zu berücksichtigen. Die Optimierung wird mit der *Python* Toolbox *scipy* und dem gradientenbasierten Optimierer SLSQP durchgeführt. Der Optimierer verwendet das neuronale Netz, um einen optimalen Temperaturverlauf zu bestimmen. Eine Optimierung dauert für jeden der vier Beispieltage weniger als eine halbe Sekunde. In den folgenden Diagrammen werden gegenübergestellt:

- Blaue Kurve: Initialer Ausgangszustand (IST): Ergebnis von *SIR 3S*[®] aus einer Nachberechnung („reconstruction simulation“) basierend auf den Prozessdaten von Beispieltag 1.
- Orange Kurve: Validierung des Optimierungsergebnisses (OPT validiert): Ergebnis von *SIR 3S*[®] aus einer Nachberechnung („reconstruction simulation“) basierend auf dem Optimierungsergebnis des neuronalen Netzes.
- Grüne Kurve: In einigen Abbildungen ist zusätzlich direkt das Ergebnis der Optimierung mit dem neuronalen Netz dargestellt (NN OPT).

Abbildung 7.8a zeigt das Ergebnis der Vorlauftemperaturoptimierung. Die Temperatur wird im Vergleich zu der tatsächlich an Beispieltag 1 nahezu konstant gefahrenen Vorlauftemperatur deutlich abgesenkt. Der Verlauf der optimierten Vorlauftemperatur bildet die Ganglinie der Wärmeeinspeisung nach. Dies führt zu einer Vergleichmäßigung der Netzumwälzmenge. Die Netzumwälzmenge wird durch niedrige Vorlauftemperaturen gerade so erhöht, dass keine der oben genannten Restriktionen verletzt werden. Abbildung 7.8b zeigt den resultierenden geringeren Wärmeverlust. Die Wärmeverluste bilden den Verlauf der Vorlauftemperatur nach, da in der verwendeten quasi-stationären Berechnung keine Netzspeichereffekte berücksichtigt werden: Es besteht ein linearer Zusammenhang zwischen den berechneten Wärmeverlusten und der optimierten Vorlauftemperatur.

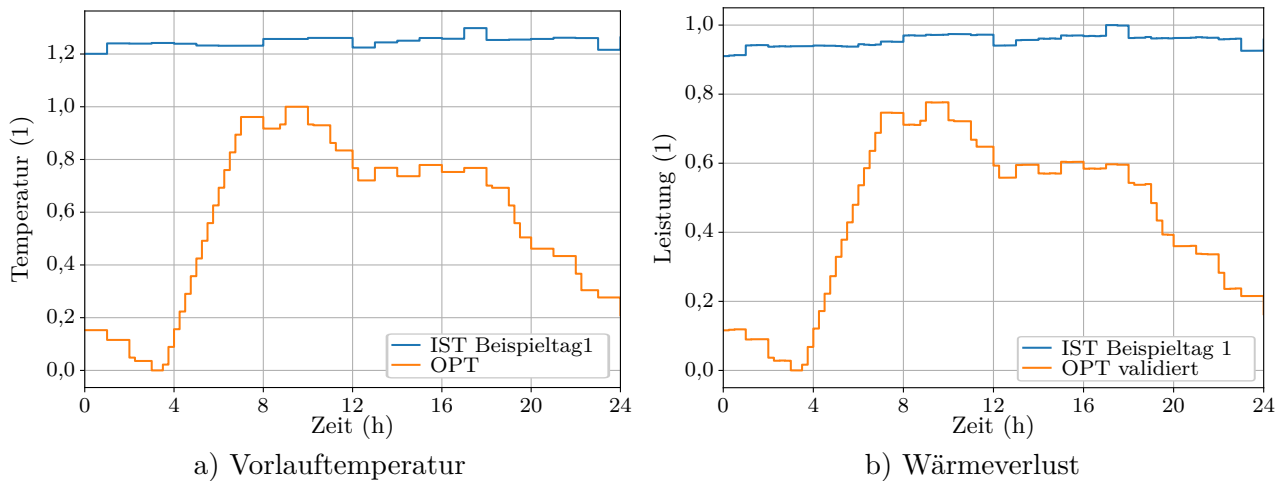


Abbildung 7.8.: Normierte Vorlauftemperaturen und Wärmeverluste an Beispieltag 1 im Januar vor (IST) und nach (OPT) der Vorlauftemperaturoptimierung.

Abbildung 7.9a zeigt den Effekt der Optimierung auf die gesamte Wärmeeinspeisung in Fernwärmenetz 1. Die Reduzierung des Wärmeverlusts ist kaum sichtbar und im Verhältnis zur gesamten Wärmeeinspeisung gering. In Abbildung 7.9b ist weiterhin der minimale Differenzdruck dargestellt. Dieser ergibt sich basierend auf einer in allen Rechnungen identischen Pumpenfahrweise, welche an acht Regelorten Soll-Differenzdrücke sicherstellt. Durch die blaue Linie (IST Beispieltag 1) wird deutlich, wie stark der minimale Differenzdruck im Tagesverlauf trotz konstanter Sollwerte an den acht Regelorten schwanken kann. Diese Schwankungen sind zurückzuführen auf signifikant steigende hydraulische Verluste bei steigenden Netzumwälzmengen entlang der Leitungsabschnitte zwischen Regelort und Verbraucher an den Netzen. Das Ergebnis des neuronalen Netzes in Abbildung 7.9b (NN OPT) entspricht zu den meisten Zeitpunkten dem als Grenzwert definierten Soll-Mindestwert (hier 0 bar). Demnach ist der minimale Differenzdruck die maßgebende Restriktion und verhindert in diesem Optimierungsfall eine weitere Vorlauftemperaturabsenkung. Der Verlauf einer mit optimierter Vorlauftemperatur durchgeführten quasi-stationären Simulation (*SIR 3S*[®] OPT validiert) zeigt, dass der finale Verlauf zu einigen Zeitpunkten leicht unter der Begrenzung liegt. Ungenauigkeiten des neuronalen Netzes für die Berechnung des minimalen Verbraucherdruckes sind der Grund. Das neuronale Netz muss bei Bedarf nachgebessert werden. Da in den Modellen sowohl im neuronalen Netz als auch in *SIR 3S*[®] bis zu einem bestimmten Maß Ungenauigkeiten zu erwarten sind, könnten die Grenzwerte auch mit Sicherheitstoleranzen belegt werden, um sicherzustellen, dass im Wirkbetrieb alle Grenzwerte mit ausreichender Sicherheit eingehalten werden. Alle weiteren Restriktionen neben dem minimalen Verbraucher Differenzdruck werden in der Optimierung eingehalten.

In Abbildung 7.9c sind die Umwälzmengen der IST und optimierten Fahrweise abgebildet. Die Netzumwälzmenge und entsprechend die Pumpendrehzahl vor und nach der Optimierung bleiben durch die Anpassung der Vorlauftemperatur an die Ganglinie der Wärmeeinspeisung relativ konstant. Der Wärmeverlust in quasi-stationären Simulationen kann mit diesem Verfahren erfolgreich reduziert werden. Allerdings steigen durch eine geringere Vorlauftemperatur die Netzumwälzmenge und damit die Leistungen der Pumpen. Der Unterschied der eingespeisten Wärmeenergie und benötigten Pumpenenergie zwischen optimierter und nicht optimierter Fahrweise ist in Abbildung 7.9d dargestellt.

Durch die reduzierte Vorlauftemperatur verringert sich die benötigte Wärmeeinspeisung um etwa 12,5 MW h. Allerdings steigt die benötigte Energie der Pumpen um circa 9,38 MW h. Die gesamte benötigte Energie sinkt folglich netto um knapp 3,12 MW h. Die aufgestellte Bilanz zwischen eingesparter Wärmeeinspeisung und Mehrenergie für die Pumpen ist unter anderem abhängig von den Wärmeverlusten. Die mit *SIR 3S*[®] berechneten Wärmeverluste basieren auf konservativen Annahmen für die Wärmeleitfähigkeit (kT -Wert) der Einzelrohrleitungen. *SIR 3S*[®] verwendet diese nennwei-

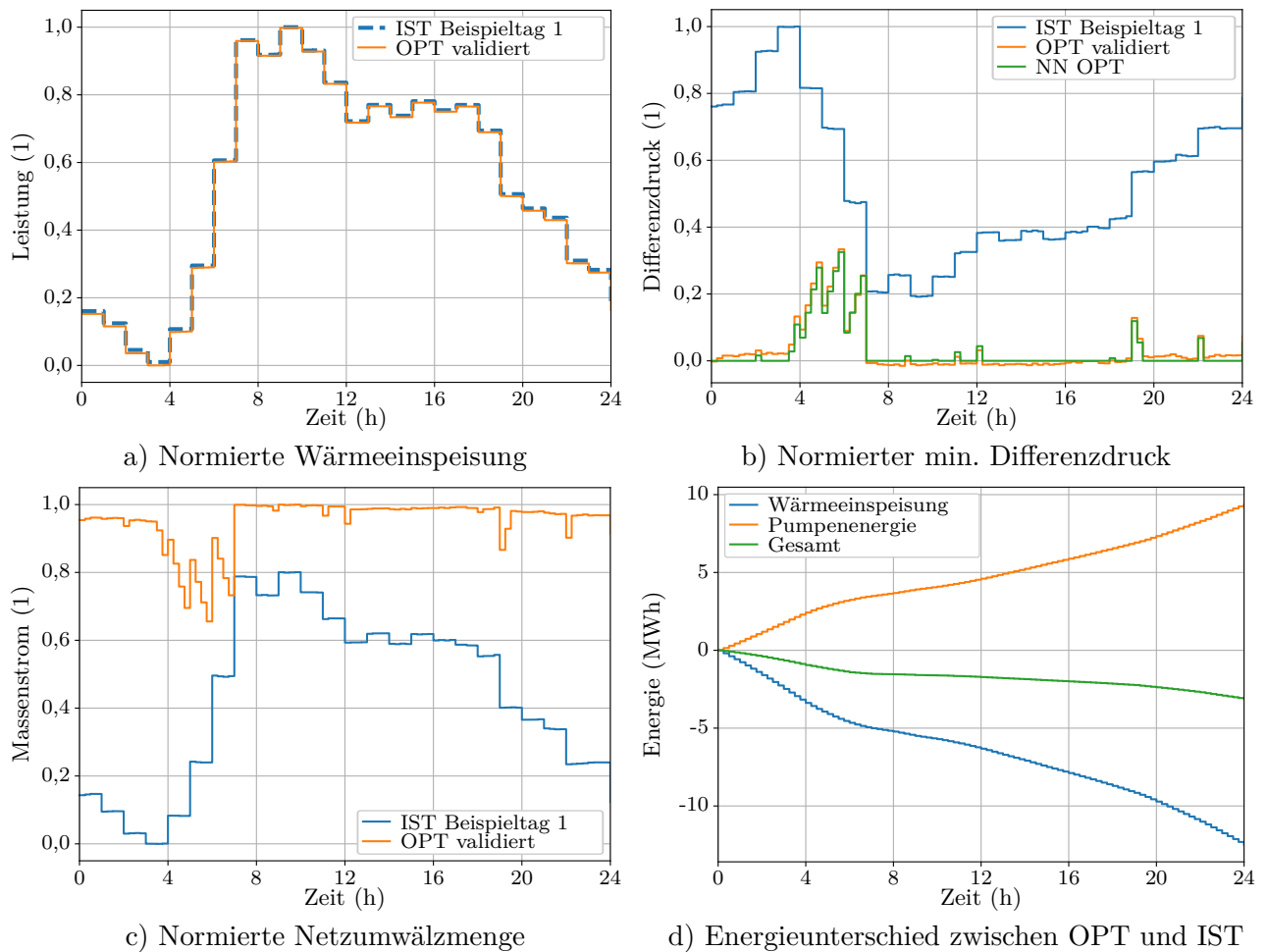


Abbildung 7.9.: Simulationsergebnisse für Beispieltag 1 im Januar vor (IST) und nach (OPT) der stationären Vorlauftemperoptimierung von Fernwärmenetz 1.

tenabhängigen kT -Werte, um mit dem Temperaturgradienten zwischen Wasser- und angenommener Außentemperatur ($10\text{ }^\circ\text{C}$) die Wärmeverluste für jede Einzelrohrleitung zu ermitteln. Die konservative Auslegung der kT -Werte äußert sich in einem prozentual gesehen niedrigen Wärmeverlust von etwa $3,8\%$ ($\hat{=}$ Wärmeverlust/Wärmeeinspeisung) für das vollständige Netzmodell an Beispieltag 1. In der Praxis können Wärmeverluste bis zu 10% oder sogar größer auftreten. Demnach wären höhere Wärmeverluste und Netto-Energieeinsparungen durch Vorlauftemperaturabsenkungen realistisch. Für eine endgültige wirtschaftliche Bewertung der Vorlauftemperaturabsenkung ist somit eine Kalibrierung der thermischen Wärmeverluste erforderlich, welche hier noch nicht durchgeführt wurde.

In der vorgestellten Optimierung wurde versucht, den Wärmeverlust in jedem Zeitschritt zu minimieren. Nun soll die Gesamtenergie minimiert werden, um auch die Pumpenleistung mit in die Optimierung einzubeziehen. Somit werden nun sowohl der Wärmeverlust als auch die Pumpenleistung in die Zielfunktion implementiert. Die Restriktionen bleiben gleich. In diesem Fall dauert eine Optimierung etwas länger, ist aber für alle Beispieltage in maximal $1,5\text{ s}$ möglich. Abbildung 7.10a zeigt das Ergebnis dieser zweiten Vorlauftemperoptimierung mit dem neuronalen Netz des aggregierten Fernwärmenetzes 1. Die Vorlauftemperatur zeigt einen ähnlichen Verlauf wie nach der Optimierung des Wärmeverlustes und liegt nur auf einem leicht höheren Niveau. Die Pumpen werden etwas weniger verwendet, wie beispielsweise in Abbildung 7.10b an dem minimalen Differenzdruck ersichtlich ist, welcher nur noch zu einem Zeitpunkt bis zum Minimum ausgereizt wird. Den Energieunterschied zur nicht optimierten Fahrweise zeigt Abbildung 7.10c. Insgesamt können etwa $3,15\text{ MWh}$ an Beispieltag

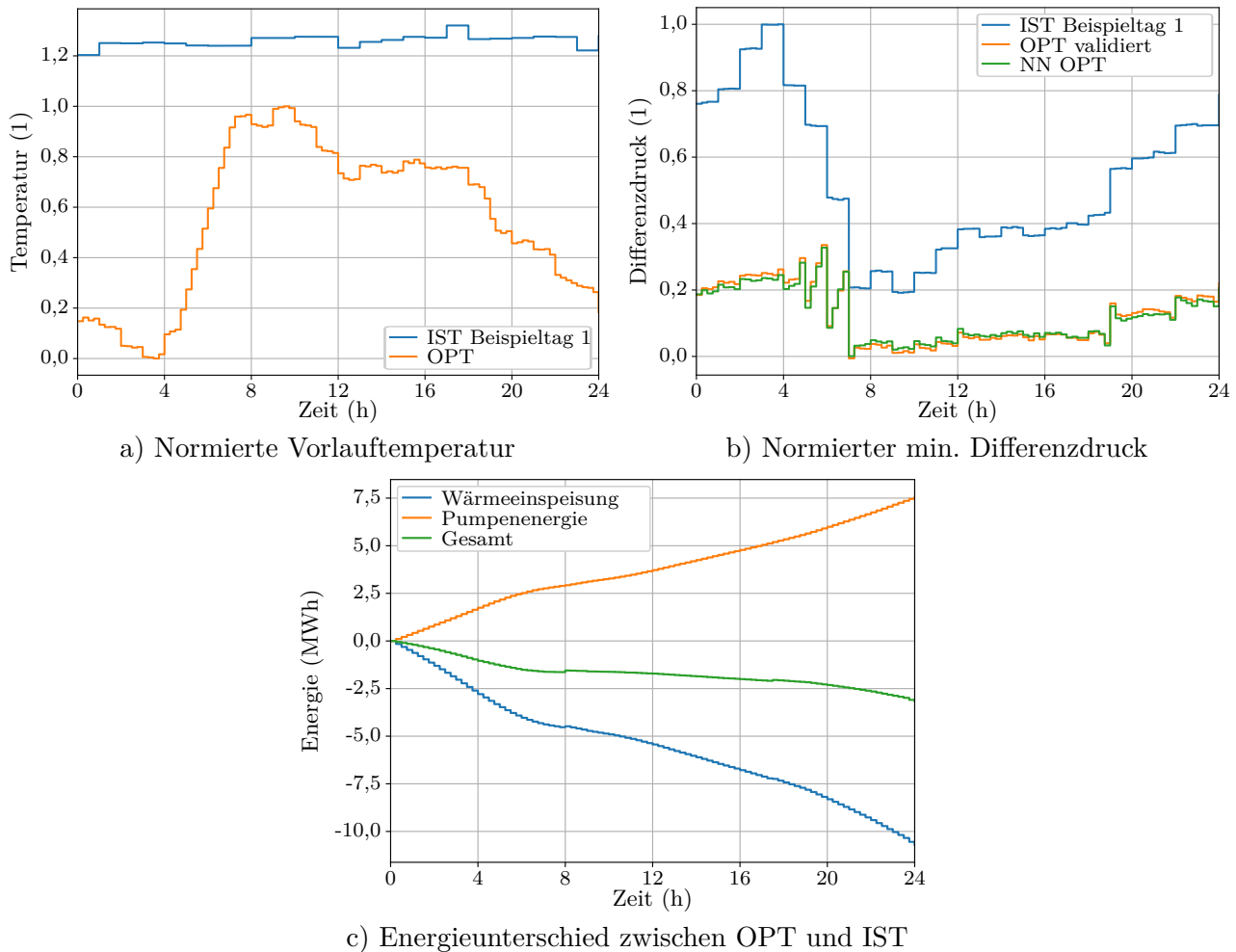


Abbildung 7.10.: Simulationsergebnisse für Beispieltag 1 im Januar vor (IST) und nach (OPT) einer Optimierung der Wärmeeinspeisung und Pumpenleistung.

tag 1 eingespart werden. Der Wärmeverlust wird um 10,71 MW h während die Pumpenenergie um 7,56 MW h steigt.

Insgesamt ist eine Optimierung mit dem Ziel der Wärmeverlust- oder Energieminimierung von Fernwärmenetz 1 also unter der Verwendung von neuronalen Netzen als stationäre Simulationsmodelle möglich. Der Ansatz ermöglicht sehr schnelle Optimierungszeiten. Die Ergebnisse sind jedoch besonders von der Kalibrierung der thermischen Wärmeverluste im Netzmodell abhängig, welche noch nicht durchgeführt wurde. In dem bisherigen Modellstatus sind nur geringe Einsparungen zu erwarten. Bei Wärmeverlusten bis zu 3,8 % ergibt sich eine Einsparung von etwa 3,15 MW h an einem Wintertag.

7.1.3. Netz 1: Instationäre Optimierung

In Kapitel 6.1.4 wurde gezeigt, dass instationäre Simulationen unter Verwendung von Vorlauftemperaturverläufen mit stärkeren Schwankungen über den Tagesverlauf nicht mehr genau mit stationären Simulationen übereinstimmen. Somit könnten nach einer Optimierung, welche wie im vorherigen Kapitel 7.1.2 ein stationäres Modell verwendet, Grenzwerte über- oder unterschritten werden. Zudem sind die berechneten Energieeinsparungen in diesem Fall anzuzweifeln. Hier soll deshalb eine Optimierung unter Verwendung instationärer Netzmodelle untersucht werden. In Kapitel 6.1.4 wurde die Systemidentifikation als Möglichkeit der Approximation instationärer Modelle vorgestellt und vorrangig nlarx-Modelle zeigen gute Ergebnisse. Auch wenn diese nicht tiefgehend untersucht wurden, kann

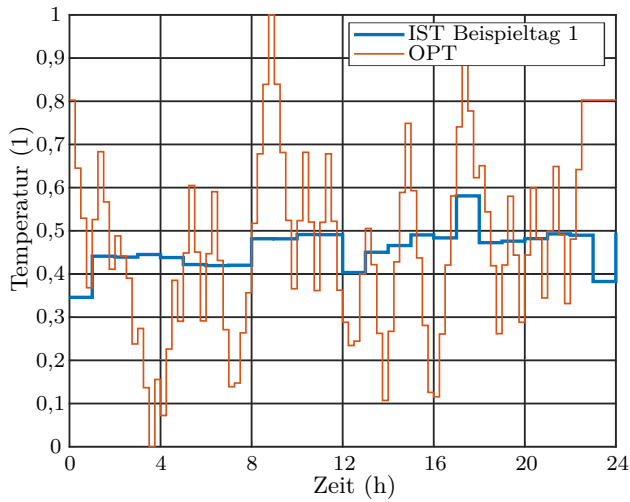
eine instationäre Optimierung mit diesen Modellen prinzipiell getestet werden. Zudem können die Systemidentifikationsmodelle auf zusätzliche Berechnungsfälle angewandt und so weitergehend validiert werden.

Ziel der Optimierung ist wieder die Vorlauftemperatur so zu wählen, dass die gesamte ins System eingespeiste Energie minimiert wird. Wärmeenergie und Pumpenenergie werden so insgesamt reduziert. Die Optimierung erfolgt über einen Zeithorizont von 24 Stunden mit Zeitschritten von einer Minute. Alle Zeitschritte werden dabei gleichzeitig optimiert, es entsteht also ein Variablenvektor von 1440 Vorlauftemperaturen. Für jeden Zeitschritt werden die gesamt eingespeiste Wärme- und Pumpenenergie vorhergesagt. Als Restriktionen für die Optimierung werden zunächst nur die ebenfalls zu berechnende maximale Strömungsgeschwindigkeit und ein maximaler Temperaturgradient gewählt. Die anderen Restriktionen werden noch nicht beachtet. Allerdings wird als weitere Restriktion eingeführt, dass die Vorlauftemperatur in den letzten zwei Stunden der Tagessimulation der initialen Temperatur entsprechen muss. Das Netz wird zum Ende des Tages also wieder in einen annähernd stationären Zustand gebracht. Wäre dies nicht der Fall, könnte die Temperatur zum Tagesende stark abgesenkt werden. Dies würde zwar die insgesamt benötigte Energie für das Netz und somit die Optimierungsergebnisse verbessern, würde aber auch den Verlauf für den nächsten Tag verschlechtern. Insgesamt müssen von der Optimierung also 1440 Variablen optimiert werden. Für jede Auswertung der Zielfunktion werden 1440 Wärme- und Pumpenenergien mit zwei separaten Modellen vorhergesagt, für jede Auswertung der Restriktion 1440 Strömungsgeschwindigkeiten. Die Optimierung wird in *MATLAB*[®] mit dem auf dem SQP-Ansatz basierten Optimierer WORHP durchgeführt. In den folgenden Diagrammen werden gegenübergestellt:

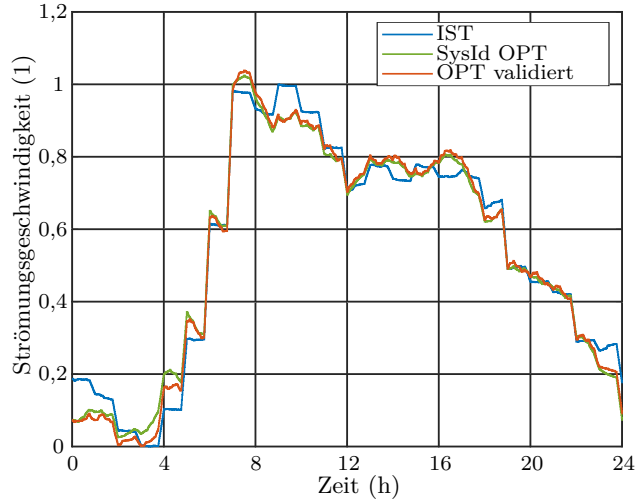
- Blaue Kurve: Initialer Ausgangszustand (IST): Ergebnis von *SIR 3S*[®] aus einer instationären Nachberechnung („reconstruction simulation“) basierend auf den Prozessdaten von Beispieltag 1.
- Orange Kurve: Validierung des Optimierungsergebnisses (OPT validiert): Ergebnis von *SIR 3S*[®] aus einer instationären Nachberechnung („reconstruction simulation“) basierend auf dem Optimierungsergebnis mit Systemidentifikation.
- Grüne Kurve: In einigen Abbildungen ist zusätzlich direkt das Ergebnis der Optimierung mit Systemidentifikation dargestellt (SysId OPT).

Abbildung 7.11a zeigt die optimierte Vorlauftemperatur, welche nur leicht um die tatsächlich gefahrene Vorlauftemperatur von Beispieltag 1 schwingt. Abbildung 7.11b zeigt den Verlauf der in der Optimierung begrenzten maximalen Strömungsgeschwindigkeit. Der Grenzwert von 1,3 m/s (mit Offset) wird eingehalten. Zudem ist der Verlauf des Systemidentifikationsmodells im Vergleich mit der instationären *SIR 3S*[®] Simulation zu sehen. Die Modelle weichen nur geringfügig voneinander ab mit einer maximalen Abweichung unter 0,05 m/s. In Abbildung 7.11c ist der Verlauf der minimalen Differenzdrücke dargestellt. Mit der optimierten Vorlauftemperatur sind der Änderung der Temperatur entsprechende Schwingungen zu sehen, welche zu einem Zeitpunkt auch den Grenzwert von 0,1 bar (mit Versatz der y-Achse) unterschreiten.

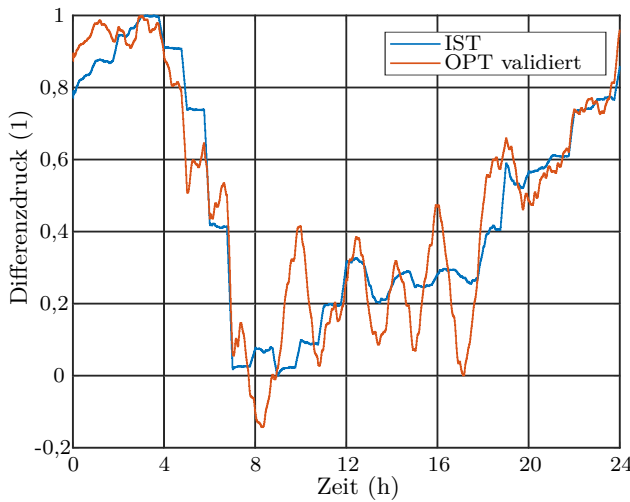
Die Abbildungen 7.11d und 7.11e zeigen jeweils die initiale und optimierte Wärmeeinspeisung und Pumpenleistung. Zum Vergleich ist das Ergebnis der Systemidentifikation dargestellt. Das Modell der Wärmeeinspeisung stimmt sehr gut mit dem *SIR 3S*[®] Modell überein. Das Modell der Pumpenleistung zeigt keine ausreichende Übereinstimmung und sollte nachgebessert werden. Das Resultat der Optimierung im Sinne eingesparter Energie zeigt Abbildung 7.11f. Insgesamt können etwa 2,3 MWh eingespart werden. Allerdings ist auch ersichtlich, dass selbst die geringen Abweichungen im Modell der Wärmeeinspeisung zu größeren Abweichungen in den Ergebnissen führen. Das Systemidentifikationsmodell berechnet die Einsparung mit 4,2 MWh deutlich höher. Für eine genaue Optimierung sind also sehr genaue Modelle notwendig. Ebenfalls ist für die Systemidentifikationsmodelle anzumerken, dass sie stark abweichende Ergebnisse liefern, wenn der Bereich der für die Identifikation verwendeten



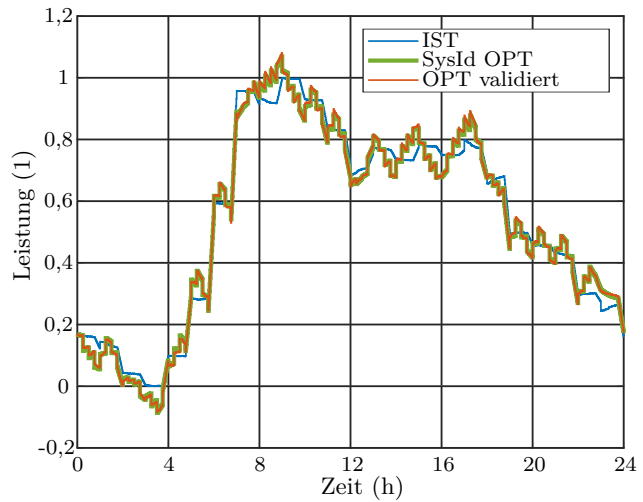
a) Vorlauftemperatur



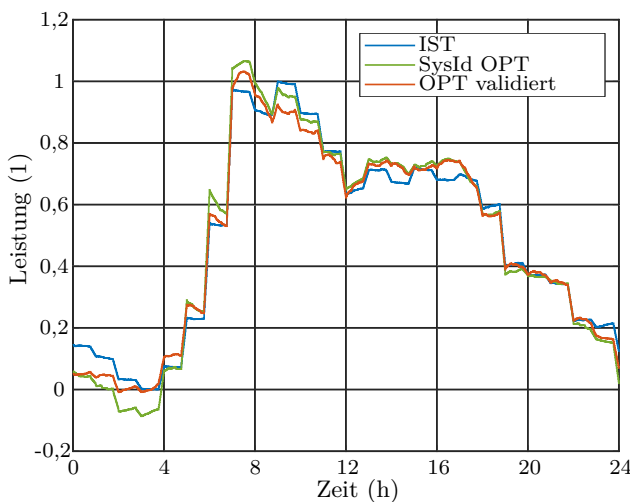
b) Maximale Strömungsgeschwindigkeit



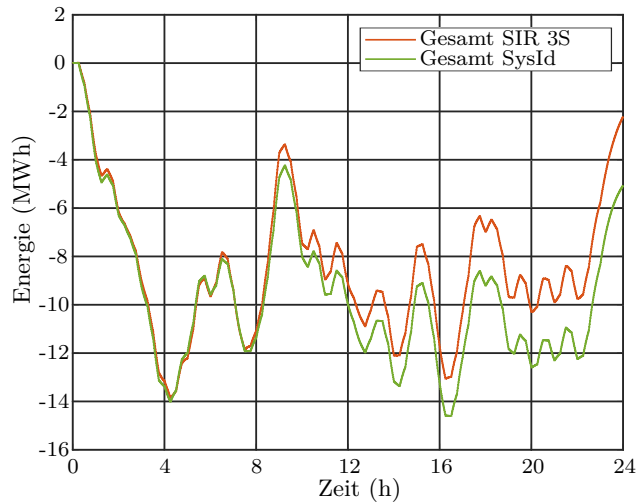
c) Minimaler Differenzdruck



d) Wärmeeinspeisung



e) Pumpenleistung



f) Energieunterschied zwischen OPT und IST

Abbildung 7.11.: Normierte Simulationsergebnisse für Beispieltag 1 im Januar vor (IST) und nach (OPT) der instationären Vorlauftemperaturoptimierung.

Eingangsdaten verlassen wird. Die identifizierten Modelle sind zum Beispiel nur für Wintertage mit hohem Wärmebedarf gültig.

Der Wärmeverlust ist in den *SIR 3S*[®] Modellen von Fernwärmenetz 1, wie bereits in der stationären Optimierung beschrieben, gering. Entsprechend ist das Einsparungspotenzial klein. Eine weitergehende Optimierung könnte die Minimierung der Kosten untersuchen, also eine Abwägung zwischen Stromkosten für die Pumpen und Kosten für die Wärmeeinspeisung ins Netz. Des Weiteren dauert die Optimierung in der beschriebenen Form etwa 1700 s und ist somit für eine praktische Anwendung zu langsam. Um kürzere Rechenzeiten zu erreichen kann entweder der Prädiktionshorizont reduziert oder eine weitere Verbesserung der Optimierung untersucht werden. Die aus der Optimierung resultierende Vorlauftemperatur schwankt um die konstante Initialfahrweise und unterscheidet sich deutlich von den Ergebnissen der stationären Optimierung in Kapitel 7.1.2. Da die instationäre Optimierung mit 1440 Variablen für alle Zeitpunkte gleichzeitig durchgeführt wird, findet diese vermutlich kein globales Optimum und erreicht nur leicht von der Initialfahrweise abweichende lokale Optima. Insgesamt kann keine einsatz- und funktionsfähige Optimierung vorgestellt werden. Lediglich das Konzept der Optimierung unter Verwendung von Systemidentifikationsmodellen konnte getestet werden und ist prinzipiell eine Möglichkeit für instationäre Optimierungen. Zusätzlich konnte gezeigt werden, dass die in Kapitel 6.1.4 identifizierten nlarx-Modelle für die Wärmeeinspeisung und die maximale Strömungsgeschwindigkeit außerhalb der Trainings- und Testdaten gute Ergebnisse im Vergleich zu *SIR 3S*[®] bei geringen Rechenzeiten erzielen.

7.2. Optimierung von Fernwärmenetz 2

In Kapitel 7.1.2 wurden neuronale Netze erfolgreich als stationäre Simulationsmodelle für die Berechnung der Restriktionen einer Optimierung verwendet. Nun kann das Ziel der optimierten verteilten Wärmeeinspeisung anhand eines einfachen Beispiels mit zwei Wärmeerzeugungsanlagen untersucht werden. Dabei kommen die in Kapitel 6.2.2 vorgestellten neuronalen Netz für quasi-stationäre Simulationen des *SIR 3S*[®] Modells vom aggregierten Fernwärmenetz 2 zum Einsatz. Durch die Optimierung sollen von der wirtschaftlichen Optimierung vorgegebene Sollwerte für Leistungen und Vorlauftemperaturen möglichst genau unter Einhaltung der folgenden, teils angenommenen Nebenbedingungen (Restriktionen) erreicht werden:

- Nenndrehzahl als maximale Drehzahl für alle Pumpen (n_{Pumpe})
- Maximale Leistungen (P) des HW und des HKW
- Maximaler Massenstrom (\dot{m}) des HW und des HKW
- Maximale Strömungsgeschwindigkeit (v) an den Wärmeerzeugungsanlagen und der DEA
- Minimaler Differenzdruck (Δp) an allen Kundenanlagen
- Minimale Vorlauftemperatur (T_{VL}) von 90 °C an allen Kundenanlagen an Beispieltag 1. Für wärmere Tage muss diese Temperatur abgesenkt werden, da niedrigere Temperaturen in das Netz gespeist werden.
- Netzweiter minimaler und maximaler Rücklaufdruck (p_{RL})
- Netzweiter minimaler und maximaler Vorlaufdruck (p_{VL})
- Maximale Netzeinspeisetemperaturgradienten am Ausgang der Wärmeerzeugungsanlagen
- Außentemperaturabhängige Vorlauftemperaturgrenzen für die beiden Wärmeerzeugungsanlagen

Die verwendeten Restriktionen basieren in Teilen auf Angaben des Netzbetreibers. Für einige Grenzwerte ohne vorliegende Angaben werden eigene Annahmen getroffen, um eine vollständige Optimierung zu erproben. Weitere Restriktionen können bei Bedarf in die Modelle und die Optimierung aufgenommen werden. Als Variablen stehen der Optimierung die Leistungen und Vorlauftemperaturen des HW sowie die Vorlauftemperatur des HKW zur Verfügung. Die Leistung des HKW ergibt sich aus der Leistungsbilanz der vorzugebenden Gesamtleistung, der Leistung des HW und der Wärmeverluste. Die DEA wird für die Optimierung nicht verwendet, da diese an den Beispieltagen nicht zum Einsatz kam.

In Kapitel 3.3.1 wurden die *Unit-Commitment*- und *Economic-Dispatch*-Probleme beschrieben. Das *Unit-Commitment*-Problem bestimmt die An- und Aus-Zustände der Wärmeerzeugungsanlagen und wird als von der wirtschaftlichen Optimierung gelöst betrachtet. Ist der Sollwert des HW also zum Beispiel 0, wird dieses nicht in der Optimierung berücksichtigt und auf 0 gesetzt. Das *Economic-Dispatch*-Problem bestimmt, welche Leistung die Wärmeerzeugungsanlagen einspeisen sollen und wird ebenfalls von der wirtschaftlichen Optimierung gelöst. So könnten allerdings die oben genannten Restriktionen verletzt werden, da sie in der wirtschaftlichen Optimierung nicht berücksichtigt werden. Die durchzuführende Optimierung verwendet also die Ergebnisse des *Economic-Dispatch*-Problems als Sollwerte und verändert sie nur so, dass alle physikalischen Grenzwerte eingehalten werden.

Die Umsetzung der Optimierung wird in Kapitel 7.2.1 vorgestellt. Für die Netzsimulation werden die in Kapitel 6.2.2 beschriebenen neuronalen Netze mit und ohne Rücklauftemperatur als Eingang getestet. Bei der Optimierung mit Rücklauftemperaturvorgabe wird die Temperatur fix entsprechend den Beispieltage vorgegeben. Zusätzlich soll eine Optimierung mit Vorhersage der Rücklauftemperatur untersucht werden. Die Optimierungen werden mit der *Python* Toolbox *scipy* mit lokalen und globalen Optimierungsansätzen für alle drei Beispieltage durchgeführt und die Ergebnisse verglichen. Der gradientenbasierte Optimierer *minimize* mit der Methode SLSQP und mehrere globale Optimierer kommen dafür zum Einsatz. Zuletzt werden instationäre Simulationen durchgeführt, um die Ergebnisse der stationären Optimierungen zu validieren.

In Kapitel 7.2.2 wird anschließend überprüft, ob die Optimierung auch mit von den realen Wärmeabnahmen abweichenden Prognosen Lösungen findet. Zudem untersuchen die Kapitel 7.2.3 und 7.2.4 die Verwendung der Vorlaufleitungen als Temperaturspeicher und eine mögliche Minimierung der Wärmeverluste.

7.2.1. Optimierung von Netz 2 nach Leistungsprognose und wirtschaftlichen Sollwerten

In einem ersten Schritt werden eine 24-Stunden-Leistungsprognose für Beispieltag 1 und die wirtschaftlich optimierten Leistungen und Vorlauftemperaturen als Sollwerte für eine Optimierung verwendet. Abweichungen bei der Leistungsprognose werden nicht berücksichtigt. Es handelt sich also um einen idealen Fall mit perfekter Prognose und entsprechend wirtschaftlich optimierten Sollwerten. Vor einer Optimierung, soll noch geprüft werden, ob bei Vorgabe von Leistungsprognose, wirtschaftlich optimierten Leistungen und Vorlauftemperaturen alle Restriktionen eingehalten werden. Dafür wird direkt eine Simulation mit der Prognose und den Sollwerten durchgeführt. Das neuronale Netz hat die gesamte Wärmeleistung, die Leistungen der Wärmeerzeugungsanlagen und die Vorlauftemperaturen als Eingänge und berechnet die Massenströme der Anlagen. Diese können anschließend für eine stationäre Simulation in *SIR 3S*[®] verwendet werden. Die Ergebnisse der Simulationen des neuronalen Netzes und von *SIR 3S*[®] für die gesamte Wärmeeinspeisung und die Leistung des HKW zeigen die Abbildungen 7.12a und 7.12b. Abbildung 7.12c zeigt zudem die Leistung des HW. Die prognostizierte Wärmeeinspeisung und die Sollwerte für die Leistungen der Wärmeerzeugungsanlagen werden in der Simulation genau getroffen. In Abbildung 7.12d ist weiterhin der minimale Rücklaufdruck im Fernwärmenetz dargestellt. Dieser unterschreitet zu einigen Zeitpunkten den minimal zulässigen Druck, welcher in der normierten Darstellung bei 0,0 liegt.

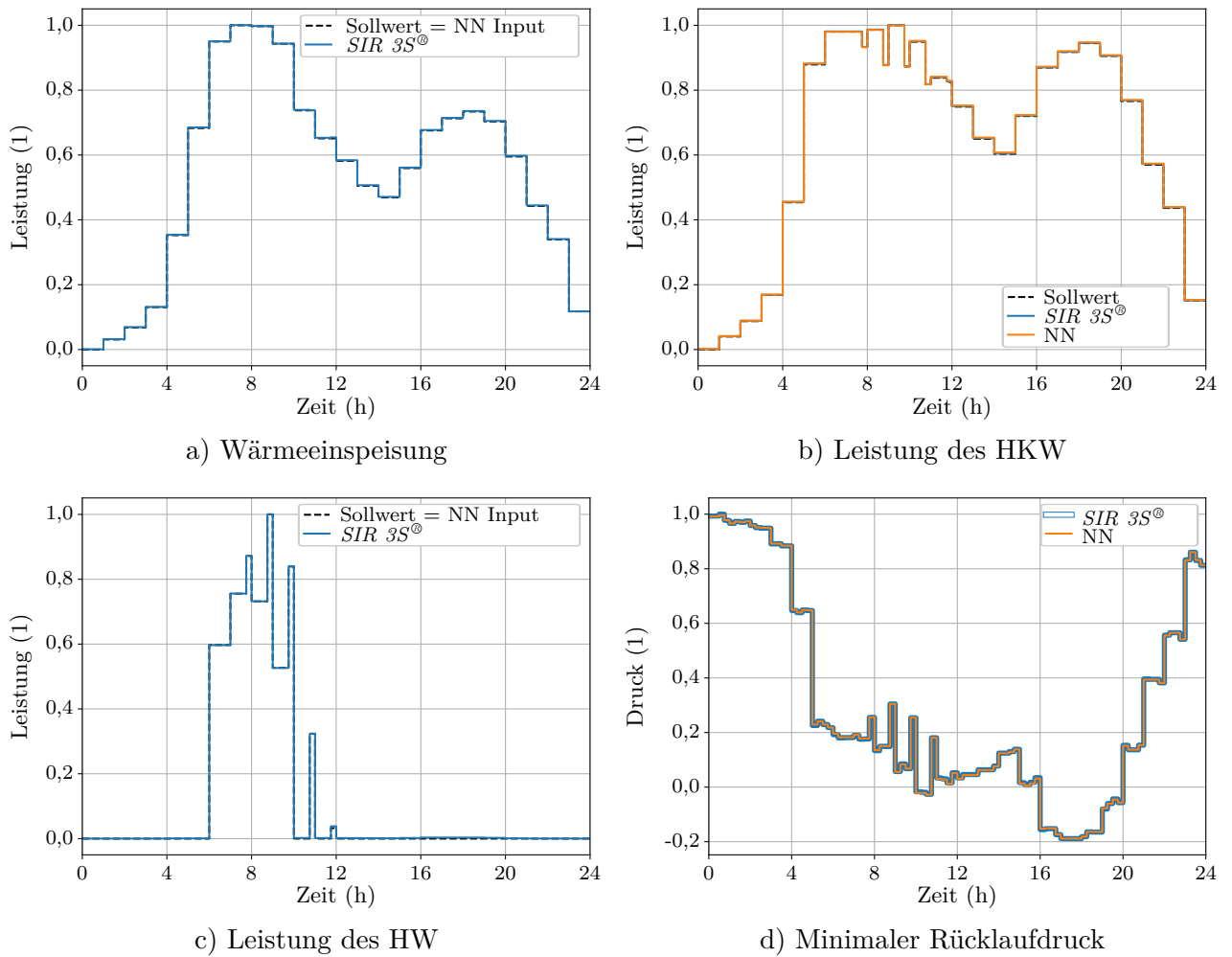


Abbildung 7.12.: Normierte Simulationsergebnisse mit wirtschaftlich optimalen Sollwerten für Fernwärmenetz 2 an Beispieltag 1 im Februar.

Die direkte Anwendung der prognostizierten Leistung und wirtschaftlich optimierten Sollwerte ergibt also zu einigen Zeitpunkten unzulässige hydraulische Zustände, welche durch eine Änderung der Lastverteilung zwischen den beiden Wärmeerzeugungsanlagen oder eine Änderung der Vorlauftemperaturen behoben werden können. Im Folgenden wird nun eine Optimierung mit diesem Ziel für 24 Stunden im 15-Minuten-Takt durchgeführt. Bei der Optimierung sollen alle Restriktionen eingehalten und Gleichung (7.23) als Zielfunktion verwendet werden. Das neuronale Netz wird in der Optimierung für die Berechnung der Restriktionen und der Zielfunktion verwendet. Anschließend werden die Ergebnisse einer Simulation mit dem neuronalen Netz und den optimierten Sollwerten als Eingänge für eine Validierungsrechnung an *SIR 3S*[®] übergeben. Die Ein- und Ausgänge der Optimierung sowie das Vorgehen sind in Abbildung 7.13 schematisch zusammengefasst.

$$\begin{aligned}
 \Delta P_{HKW} &= 100 \cdot (P_{HKW} - SW_{P_{HKW}}) \\
 \Delta P_{HW} &= 10 \cdot (P_{HW} - SW_{P_{HW}}) \\
 \Delta T_{VL, HKW} &= 10 \cdot (T_{VL, HKW} - SW_{T_{VL, HKW}}) \\
 \Delta T_{VL, HW} &= T_{VL, HW} - SW_{T_{VL, HW}} \\
 \Delta p_{PRL, min} &= 10 \cdot (p_{PRL, min} - p_{min}) \\
 f &= 0,5 \cdot (\Delta P_{HKW}^2 + \Delta P_{HW}^2 + \Delta T_{VL, HW}^2 + \Delta T_{VL, HW}^2 + \Delta p_{PRL, min}^2) \quad (7.23)
 \end{aligned}$$

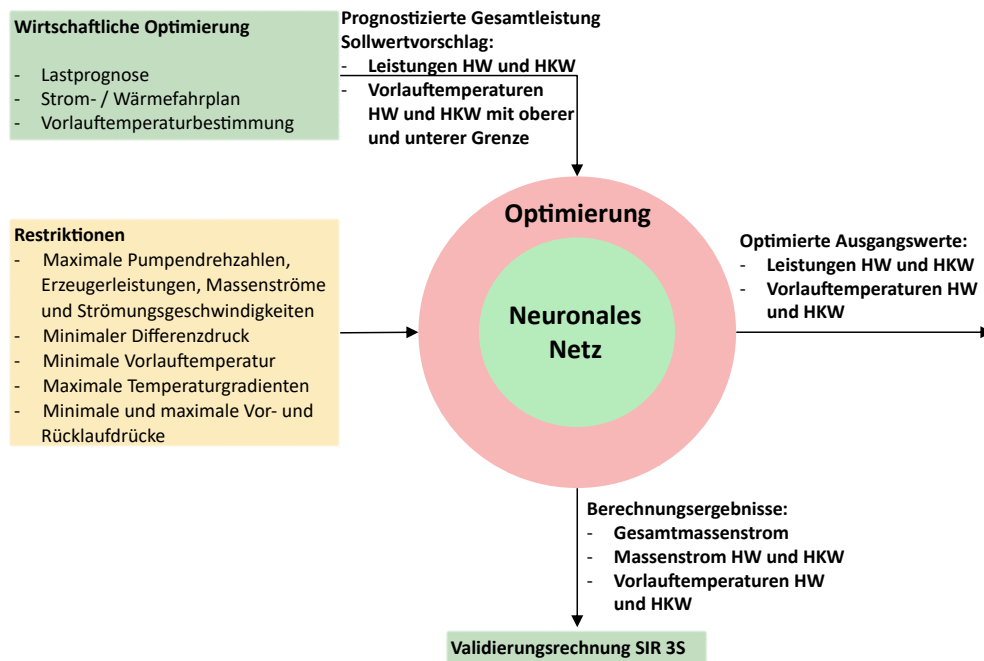


Abbildung 7.13.: Schematische Darstellung der Optimierung von Fernwärmenetz 2.

Die Zielfunktion (7.23) besteht jeweils aus der Differenz zwischen den in der Optimierung verwendeten Variablen Leistung des HW P_{HW} , Vorlauftemperaturen $T_{VL,HWK}$ und $T_{VL,HW}$ und deren Sollwerten, welche mit SW gekennzeichnet sind. Zusätzlich werden in der Zielfunktion die Berechnungsergebnisse des neuronalen Netzes für die Leistung des HKW P_{HKW} und der minimale Druck im Rücklauf $p_{RL,min}$ verwendet. Die Leistung des HKW soll wieder möglichst dem Sollwert entsprechen, während der minimale Rücklaufdruck einen minimal vorgegebenen Druck p_{min} erreichen soll. So wird die Druckspreizung im Netz möglichst gut ausgenutzt. Des Weiteren werden einige der Komponenten der Zielfunktion mit Gewichtungsfaktoren versehen. Die Leistung des HKW ist zum Beispiel mit 100 am höchsten gewichtet, da dieses Kraftwerk auch für die Stromproduktion relevant ist und die wirtschaftliche Optimierung zuvor entsprechend durchgeführt wurde. Die Gewichtungsfaktoren wurden variiert und haben mit den dargestellten Werten zu guten Ergebnissen geführt. Zuletzt werden alle einzelnen Ziele zur Zielfunktion addiert. Die Gewichtungsfaktoren und der Aufbau der Zielfunktion können in dem Optimierungscodex bei Bedarf angepasst werden. Die Ergebnisse sind in den folgenden Abbildungen wie folgt gegenübergestellt:

- Schwarz gestrichelt: Prognose / Sollwerte für Beispieltag 1. Die Prognose und wirtschaftliche Optimierung wurden am Vortag für die gesamten 24 h angefertigt.
- Orange Kurve: Validierung des Optimierungsergebnisses (OPT validiert): Ergebnis von *SIR 3S*[®] aus einer Nachberechnung („*reconstruction simulation*“) basierend auf dem Optimierungsergebnis des neuronalen Netzes.
- Grüne Kurve: In einigen Kurven ist direkt das Optimierungsergebnis des neuronalen Netzes (NN OPT) dargestellt.

Abbildung 7.14a zeigt, dass die Leistungsprognose nach der Optimierungsrechnung genau getroffen wird. Die geforderte Leistung wird ins Netz eingespeist. Zudem ist in Abbildung 7.14b der minimale Rücklaufdruck dargestellt. Zu den frühen Morgen- und Abendstunden ist keine große Druckspreizung im Netz notwendig, während den Tag über der minimale Druck (hier bei 0,0) eingehalten wird. In den Abbildungen 7.14c und 7.14d sind die Leistungen des HKW und HW dargestellt, welche ebenfalls gut eingehalten werden. Für das HW ist in den Restriktionen der Optimierung eine minimale Leistung vorgesehen, weshalb zum Beispiel die vorgegebene Leistung um 12 Uhr deutlich überschritten wird.

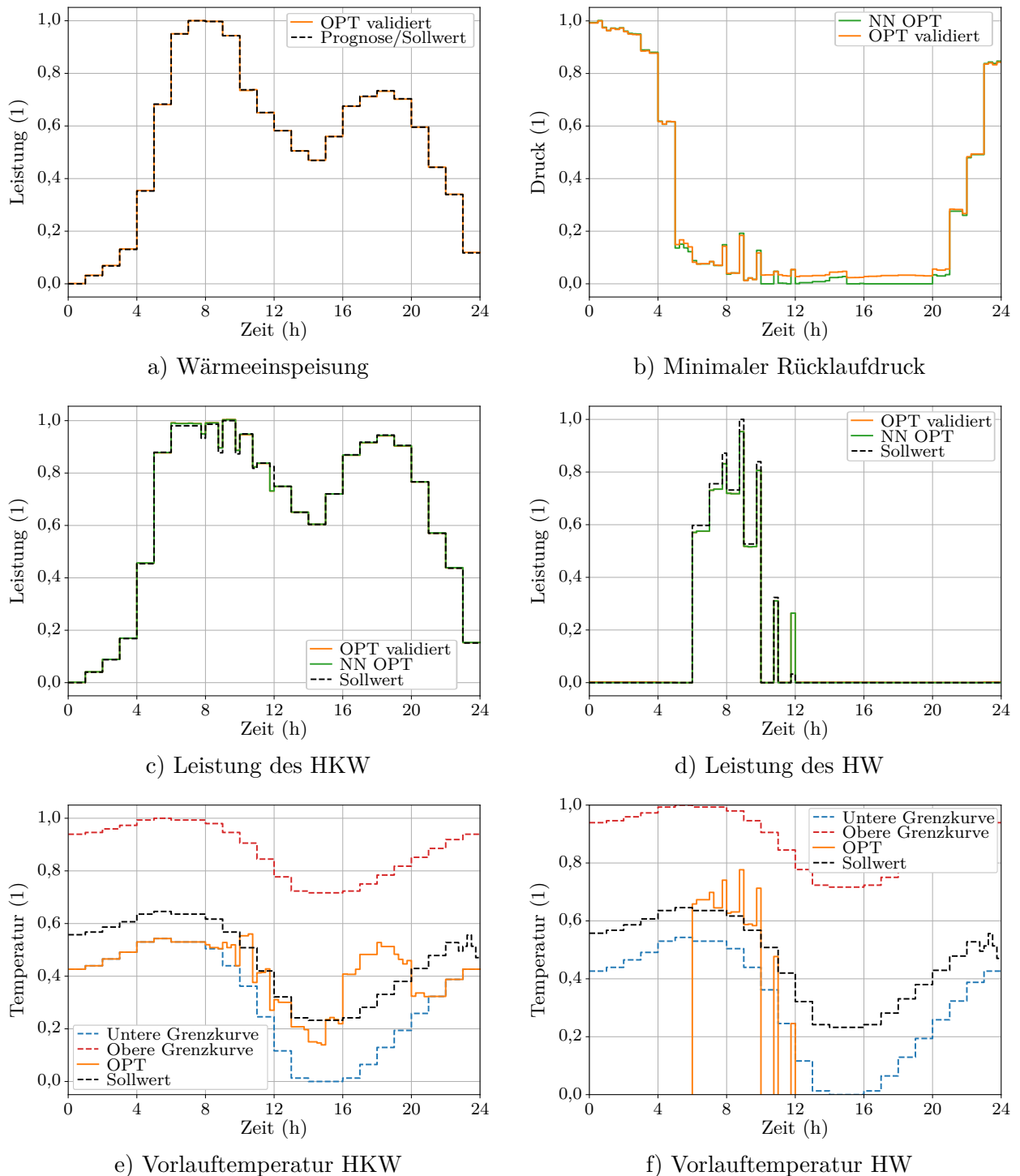


Abbildung 7.14.: Normierte Simulationsergebnisse für Beispieltag 1 im Februar nach stationärer Optimierung von Fernwärmenetz 2.

Die Vorlauftemperaturen der Kraftwerke zeigen die Abbildungen 7.14e und 7.14f. Die Vorlauftemperatur des HKW kann größtenteils unter der Solltemperatur gefahren werden und liegt nur nachmittags etwas über dem Sollwert. Für das HW ist im Aus-Zustand eine konstante niedrige Temperatur vorgegeben, im An-Zustand schwankt die Vorlauftemperatur leicht um den Sollwert.

Die vorgestellte Optimierung kann die Leistungen zwischen den zwei Wärmeerzeugungsanlagen verteilen und dabei sicherstellen, dass alle Restriktionen eingehalten werden. Die Optimierung wurde mit dem SQP-basierten Optimierer *minimize* der *Python*-Bibliothek *scipy* durchgeführt und dauert weni-

ger als eine Sekunde. Wird ein globaler Optimierungsansatz verwendet wie *differential evolution* sind etwa sechs Sekunden für ähnliche Ergebnisse notwendig. Die gradientenbasierte Optimierung ist also ausreichend und führt schneller zum Ziel. Weitere Optimierer wie der in Kapitel 2.2.6 beschriebene *shgo* und Algorithmen der Bibliotheken *pygmo* [16] wurden ebenfalls getestet, haben aber nicht zu besseren oder schnelleren Lösungen geführt.

Bisher wurde das neuronale Netz mit der Rücklauftemperatur als Eingang verwendet, welche im Realbetrieb nicht für 24 Stunden im Voraus bekannt ist. Auch mit dem neuronalen Netz ohne Rücklauf-temperatur als Eingang ist die Optimierung erfolgreich, führt aber zu etwas höheren Abweichungen zwischen der Optimierung mit dem neuronalen Netz und der Nachrechnung mit *SIR 3S*[®] wie in Kapitel 6.2.2 dargestellt. Die maximalen Abweichungen der Leistungen bewegen sich im Bereich von 3 MW. Auch mit einer Vorhersage der Rücklauf-temperatur entsprechend Kapitel 6.2.2 ist eine Optimierung möglich. In diesem Fall wird in jedem Schritt basierend auf der prognostizierten gesamten Einspeiseleistung, der Leistung des HW sowie der beiden Vorlauf-temperaturen die Rücklauf-temperatur vorhergesagt. Die Ergebnisse sind nahezu identisch, die Optimierungsdauer erhöht sich etwas. Für die gradientenbasierte Optimierung sind in diesem Fall knapp zwei Sekunden und für die globale Optimierung etwa elf Sekunden notwendig.

Für die weiteren zwei Beispieltage kann eine Optimierung ebenfalls erfolgreich durchgeführt werden. Sowohl mit als auch ohne Rücklauf-temperaturvorgabe können alle Restriktionen eingehalten und die vorgegebenen Leistungen für das HKW und HW genau erreicht werden. Die Vorlauf-temperaturen der Wärmeerzeugungsanlagen werden dafür nur minimal verändert. Die folgende Tabelle 7.1 zeigt die Optimierungsdauer für alle Beispieltage und die beiden Optimierer SLSQP und *differential evolution*. Dabei wird jeweils die Dauer mit und ohne Rücklauf-temperaturvorgabe angegeben. Für Tag 1 ist zusätzlich die Optimierungsdauer mit Vorhersage der Rücklauf-temperatur (T_{RL}) gegeben. Diese Vorhersage kann nur für Wintertage durchgeführt werden, da das neuronale Netz für die Vorhersage nur mit Daten aus dem Winter trainiert wurde. Insgesamt führt eine lokale Optimierung zum Beispiel mit dem Optimierer SLSQP für Fernwärmenetz 2 zu guten Ergebnissen und ist deutlich schneller als mit einem globalen Ansatz wie *differential evolution*. Durch die schnellen Optimierungszeiten von unter einer Sekunde ist ein Einsatz in einer MPC denkbar, wird hier aber nicht weiter untersucht.

Tabelle 7.1.: Optimierungsdauer für Netz 2 an den drei Beispieltagen mit Leistungsvorgabe aus der Vortags-Prognose.

	Dauer mit SLSQP (s)	Dauer mit <i>differential evolution</i> (s)
Tag 1 (Februar) mit T_{RL} Vorgabe	0,63	5,65
Tag 1 (Februar) mit T_{RL} Vorhersage	1,62	10,98
Tag 1 (Februar) ohne T_{RL} Vorgabe	0,90	5,55
Tag 2 (Mai) mit T_{RL} Vorgabe	0,52	2,29
Tag 2 (Mai) ohne T_{RL} Vorgabe	0,70	2,39
Tag 3 (Juli) mit T_{RL} Vorgabe	0,71	2,04
Tag 3 (Juli) ohne T_{RL} Vorgabe	0,51	2,07

Die Optimierungsergebnisse sollen nun noch mit einer instationären Simulation in *SIR 3S*[®] überprüft werden. Dafür wird eine Simulation mit den gleichen Eingangswerten wie die der stationären Simulation (OPT validiert) mit Zeitschritten von einer Minute durchgeführt. Alle Eingangsvariablen werden stückweise konstant übergeben und somit entsprechend der stationären Optimierung nur alle 15 Minuten verändert. Die gesamte Einspeiseleistung des Netzes wird interpoliert und ändert sich so wie in der Realität kontinuierlich. Die Abbildungen 7.15a und 7.15b zeigen die Leistungen des HKW und HW nach der stationären und instationären Validierungsrechnung. Die Leistungen der beiden Simulationen stimmen gut überein. Für das HKW ist sichtbar, dass in den 15-Minuten-Zeiträumen

die Leistungen etwas voneinander abweichen. Die Unterschiede müssen im praktischen Einsatz von der unterlagerten Differenzdruckregelung der Pumpen kompensiert werden. Eine entsprechende Prüfung dieses Vorgehens steht noch aus. In den Abbildungen 7.15c und 7.15d sind zusätzlich die minimalen Differenzdrücke und Strömungsgeschwindigkeiten am HKW abgebildet, welche als Restriktionen zum Einsatz kommen. Die Strömungsgeschwindigkeit zeigt kaum Unterschiede zwischen der instationären und stationären Simulation, während der minimale Differenzdruck einige Schwingungen aufweist. Diese liegen allerdings absolut unter 0,03 bar. Insgesamt sind die Abweichungen zwischen instationärer und stationärer Simulation gering.

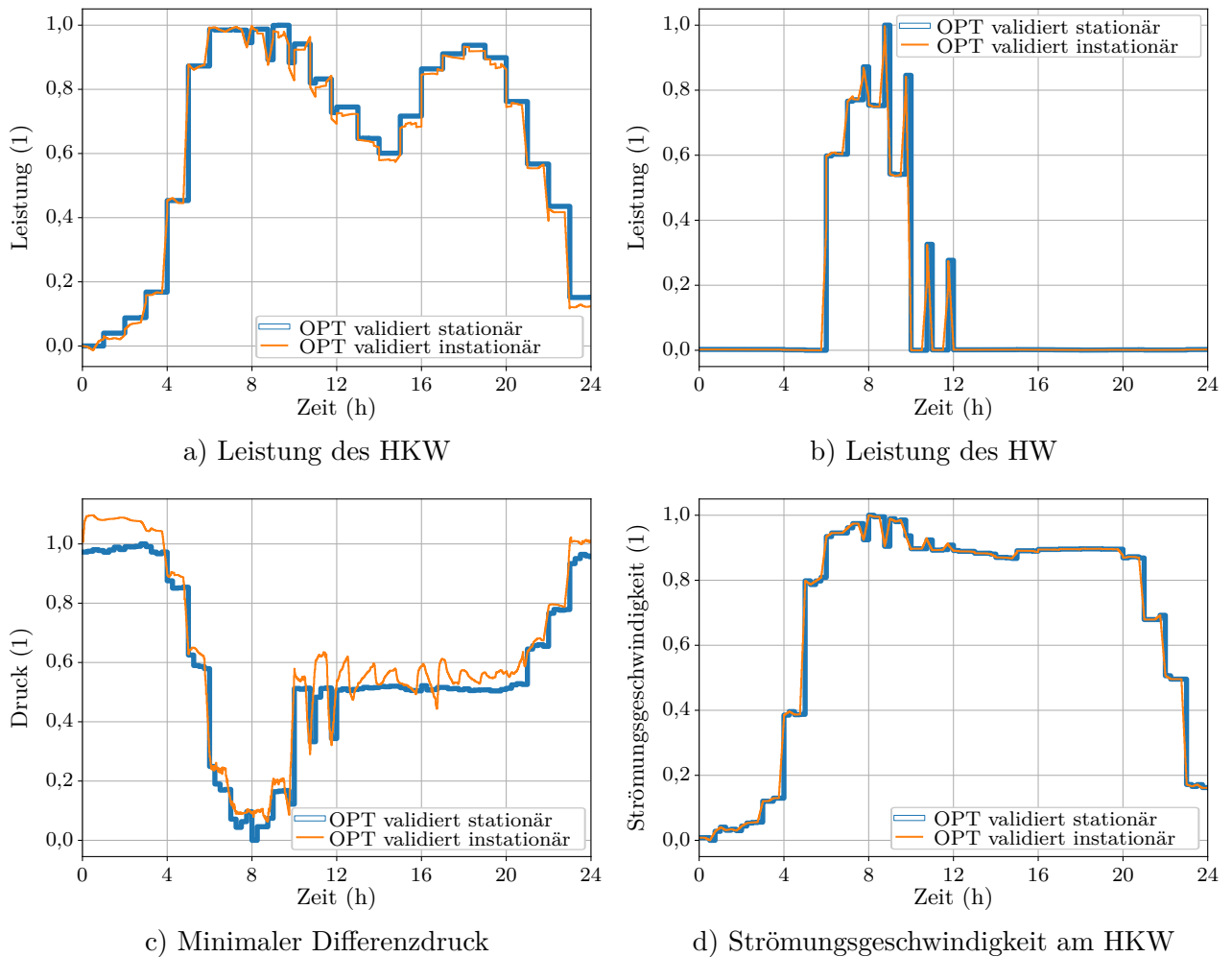


Abbildung 7.15.: Normierte stationäre und instationäre Simulationsergebnisse nach Optimierung von Fernwärmenetz 2 an Beispieltag 1 im Februar.

Insgesamt konnte in diesem Abschnitt gezeigt werden, dass eine Optimierung der verteilten Wärmeinspeisung mit dem Ziel der Einhaltung wirtschaftlich optimaler Sollwerte und unter Verwendung von neuronalen Netzen erfolgreich und schnell durchgeführt werden kann. Die Optimierung wurde auf drei Beispieltage zu unterschiedlichen Jahreszeiten angewandt. Zudem wurden verschiedene Optimierer getestet, von welchen sich der gradientenbasierte Ansatz SLSQP als beste Lösung herausstellt. Instationäre Simulationen konnten die Ergebnisse der stationären Optimierung bestätigen.

7.2.2. Optimierung von Netz 2 mit Einspeiseleistung aus Prozessdaten und wirtschaftlichen Sollwerten

Neben der Rücklauftemperaturprognose kann auch die Leistungsprognose im praktischen Einsatz fehlerbehaftet sein. Deshalb wird überprüft, wie sich die Optimierung verhält, falls die optimierten Wärmeerzeugungssollwerte nicht zu der gesamt ins Netz einzuspeisenden Leistung passen. Die Gesamtleistung zum Zeitpunkt $t = 0$ ist dabei in der Praxis messbar und kann entsprechend mit in die Optimierung übernommen werden. Auch für die restlichen 24 Stunden der Optimierung könnte so eine Abweichung der vorhergesagten Leistung von der IST-Leistung berücksichtigt werden. Als Beispiel wird die tatsächlich gemessene Leistung von Beispieltag 1 zusammen mit den wirtschaftlich optimierten Wärmeleistungen vom Vortag für eine Optimierung verwendet. Der Optimierung wird also eine Gesamtleistung vorgegeben, welche nicht zu der wirtschaftlichen Optimierung passt. Die Gesamtleistung ist höher oder niedriger als die Summe der optimierten Wärmeleistungen. In den folgenden Diagrammen werden die Ergebnisse der Optimierung wie folgt gegenübergestellt:

- Schwarz gestrichelt: Prognose / Sollwerte für Beispieltag 1. Die Prognose und wirtschaftliche Optimierung wurden am Vortag für die gesamten 24 h angefertigt.
- Blaue Kurve: Initialer Ausgangszustand (IST): Ergebnis von *SIR 3S*[®] aus einer Nachberechnung basierend auf den Prozessdaten von Beispieltag 1.
- Orange Kurve: Validierung des Optimierungsergebnisses (OPT validiert): Ergebnis von *SIR 3S*[®] aus einer Nachberechnung basierend auf dem Optimierungsergebnis des neuronalen Netzes.
- Grüne Kurve: In einigen Kurven ist direkt das Optimierungsergebnis des neuronalen Netzes (NN OPT) dargestellt.

Abbildung 7.16 zeigt zunächst die prognostizierte Einspeiseleistung an Beispieltag 1 und die tatsächlich eingespeiste Leistung aus der Nachrechnung von *SIR 3S*[®]. Hier ist ein deutlicher Unterschied sichtbar, welcher unter anderem durch Ungenauigkeiten der 24-Stunden-Prognose am Vortag erklärt werden kann. Auch das Ergebnis nach der Optimierung ist dargestellt. Die optimierte Einspeiseleistung (orange) weicht kaum von der IST-Leistung (blau) ab, welche nun als Vorgabe für die Optimierung verwendet wird.

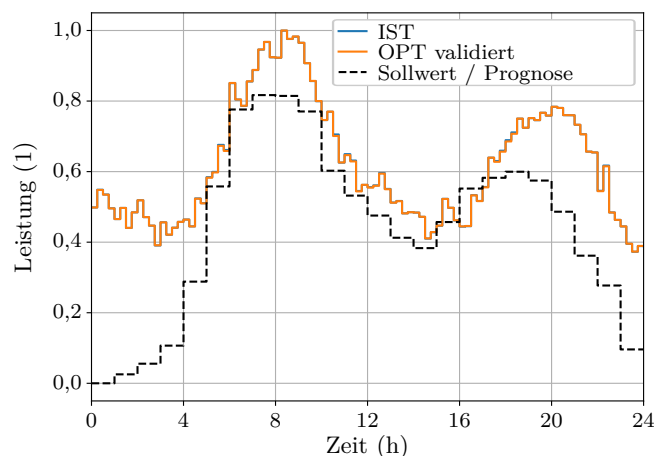


Abbildung 7.16.: Normierte gesamte Einspeiseleistung in Fernwärmenetz 2 an Beispieltag 1 (Februar).

Bisher wird das bereits erwähnte *Unit-Commitment*-Problem als gelöst vorausgesetzt. Ist der Sollwert des HW 0 MW, so wird dieses in der Optimierung nicht berücksichtigt. In diesem Fall kann bei abweichenden Leistungsprognosen allerdings nicht immer ein Optimierungsergebnis gefunden werden. Bei zu hohen gesamten Einspeiseleistungen und einem Sollwert des HW von 0 MW kann zum Beispiel der minimale Rücklaufdruck unterschritten werden. In der Optimierung wird dieser Fall abgefragt

und bei nicht erfolgreicher Optimierung mit dem Sollwert 0 wird diese wiederholt. In der zweiten Optimierung sind dem HW nur dessen Leistungsgrenzen als Restriktion vorgegeben. Die gleichen Restriktionen wie zuvor und dieselbe Zielfunktion (7.23) werden verwendet.

Die Einspeiseleistungen des HKW und des HW mit diesem Vorgehen sind in den Abbildungen 7.17a und 7.17b dargestellt. Die IST-Fahrweise von Beispieltag 1 folgt nicht den wirtschaftlich optimierten Sollwerten. Die Gründe können wieder eine abweichende Prognose oder andere durch das Personal vorgegebene Fahrweisen sein. So können zum Beispiel andere Gegebenheiten am Energiemarkt zur Umplanung führen oder die Wärmeleistungen im HKW zwischen den drei angeschlossenen Fernwärmenetzen anders verteilt werden. Bei der hier angewandten Optimierung folgt das HW größtenteils seiner Sollleistung von 0 MW. Die Differenz zur Gesamtleistung wird vom HKW geliefert. An einigen Zeitpunkten wird das HW trotz einer Sollleistung von 0 MW eingeschaltet. Andernfalls wären Restriktionen verletzt worden. Die Optimierung funktioniert daher wie vorgeschrieben.

Die Abbildungen 7.17c und 7.17d zeigen die Vorlauftemperaturen für das HKW und das HW, welche sich an der Sollwertvorgabe orientieren und zwischen unterer und oberer Grenzkurve bewegen. Vor allem am Nachmittag sind höhere Temperaturen als vorgegeben notwendig. Die IST-Vorlauftemperatur des Beispieltages zeigt teilweise Vorlauftemperaturen unter oder über den Grenzkurven. Hier ist anzumerken, dass es sich bei den Grenzkurven um Berechnungen vom Vortag handelt und sich diese während eines Tagesverlaufs ändern können. Für die Optimierung in dieser Arbeit sind nur die Grenzkurven der Vortagesoptimierung bekannt, welche somit als Grenzwerte verwendet werden.

Bei der Optimierung werden die Gradientenvorgaben und alle weiteren Restriktionen des Netzes erfolgreich eingehalten. Beispielhaft zeigen die Abbildungen 7.17e und 7.17f die netzweiten minimalen Differenz- und Rücklaufdrücke. In den Graphen ist neben den *SIR 3S*[®] Ergebnissen der IST- und OPT-Fahrweisen auch das Ergebnis der Optimierung mit dem neuronalen Netz (ML OPT) dargestellt. Der Grenzwert liegt in den normierten Graphen jeweils bei 0,0. Es ist sichtbar, dass der minimale Rücklaufdruck wieder begrenzend für die Optimierung ist und gut eingehalten wird. Nachmittags befindet sich der Rücklaufdruck längere Zeit an seinem Minimum. Abends steigt der minimale Rücklaufdruck an. Auch das HW wird trotz Aus-Vorgabe nach Lösen des *Unit-Commitment*-Problems verwendet, um den Rücklaufdruck über dem Grenzwert zu halten.

Die durchgeführte Optimierung kann also mit abweichenden Gesamtleistungen und nicht passend optimierten Einspeiseleistungen umgehen und Ergebnisse finden, welche alle Restriktionen einhalten. Getestet wird die Optimierung wieder mit lokalen (SLSQP) und globalen Optimierern (*differential evolution*) für alle Beispieltage. Sowohl das neuronale Netz mit der Rücklauftemperatur als Eingang als auch ohne diese wird angewandt. Die Dauer für die unterschiedlichen Konfigurationen und Optimierer sind in der folgenden Tabelle 7.2 aufgelistet.

Tabelle 7.2.: Optimierungsdauer für Netz 2 an den drei Beispieltagen mit Leistungsvorgabe aus Prozessdaten.

	Dauer mit SLSQP (s)	Dauer mit <i>differential evolution</i> (s)
Tag 1 (Februar) mit T_{RL} Vorgabe	17,13	18,85
Tag 1 (Februar) mit T_{RL} Vorhersage	17,94	42,49
Tag 1 (Februar) ohne T_{RL} Vorgabe	11,91	16,10
Tag 2 (Mai) mit T_{RL} Vorgabe	0,73	1,73
Tag 2 (Mai) ohne T_{RL} Vorgabe	0,71	1,72
Tag 3 (Juli) mit T_{RL} Vorgabe	0,88	2,40
Tag 3 (Juli) ohne T_{RL} Vorgabe	0,48	2,49

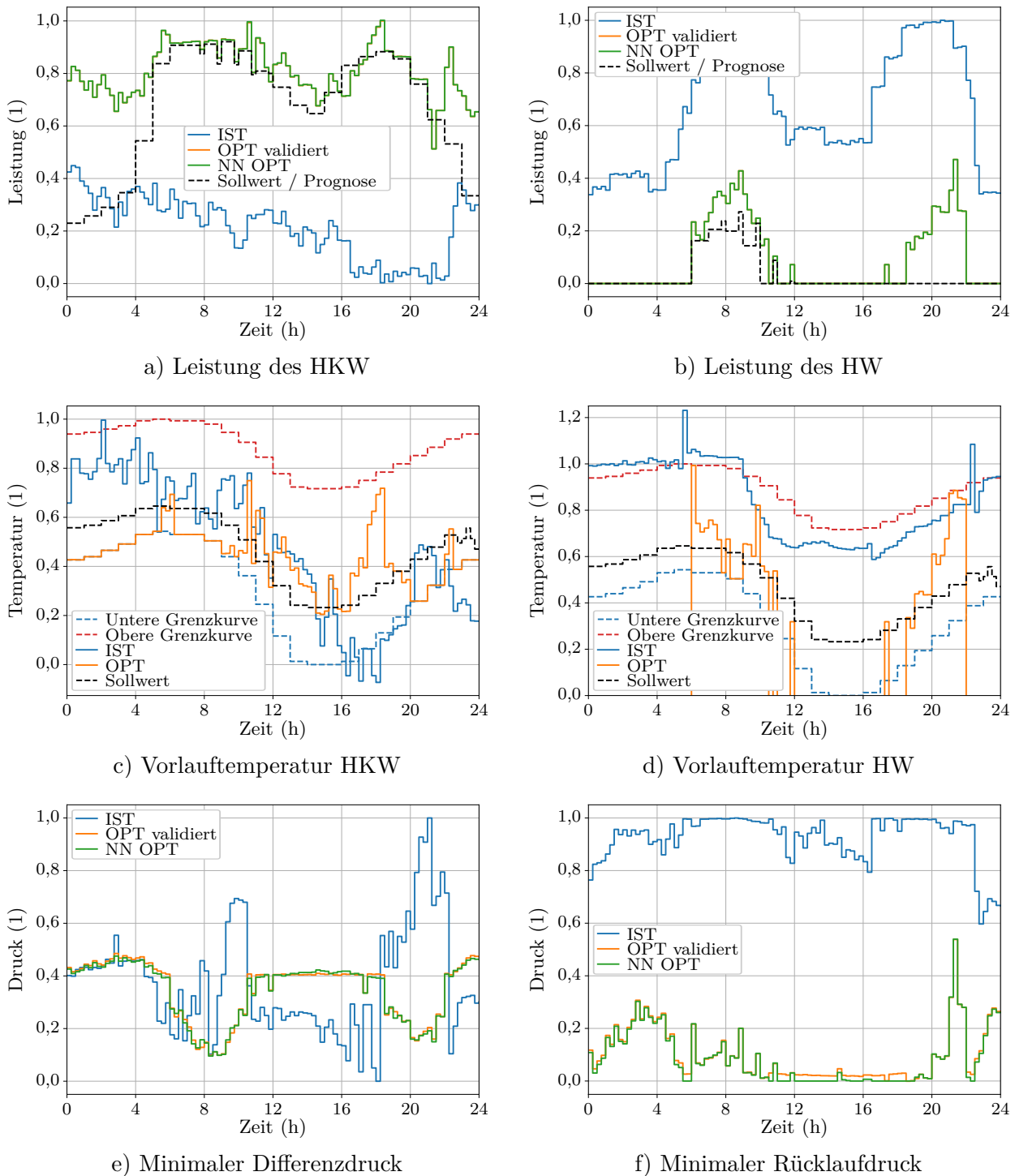


Abbildung 7.17.: Normierte Simulationsergebnisse nach Optimierung von Fernwärmenetz 2 an Beispieltag 1 im Februar mit Leistungsvorgabe aus Prozessdaten.

Für Tage 2 und 3 ist die lokale Optimierung wieder in unter einer Sekunde möglich. Für Tag 3 muss der Grenzwert des minimalen Vorlaufdrucks dabei leicht reduziert werden, da eine Lösung innerhalb der Restriktionen nicht möglich ist. Hier müsste die Druckhaltung im Modell angepasst werden. Für Tag 1 dauert die Optimierung deutlich länger und es kommt häufig zu Wiederholungen der Optimierung,

da Restriktionen mit der Aus-Vorgabe des HW nicht eingehalten werden und somit die Optimierung noch mal mit An-Vorgabe durchgeführt wird.

In den Abbildungen 7.17a und 7.17b war bereits ersichtlich, dass mit diesem Vorgehen zwar der Aus-Zustand des HW an den meisten Zeitpunkten eingehalten wird, das HKW allerdings die restliche Leistung liefern muss und so stark vom Sollwert abweicht. Da das HKW auch Strom produziert, ist unter Umständen eine genaue Einhaltung der vorgegebenen Heizleistung eher optimal. Der Optimierung kann deshalb wahlweise ein permanenter An-Zustand des HW vorgegeben werden und dieses somit für die zusätzlich benötigte Leistung verwendet werden. In diesem Fall werden die Ergebnisse des *Unit-Commitment*-Problems der wirtschaftlichen Optimierung nicht beachtet. Abbildung 7.18 zeigt das Ergebnis der Wärmeerzeugungsleistungen einer solchen Optimierung. Die Sollleistung des HKW wird nun deutlich genauer eingehalten. Zu einigen Zeitpunkten wird die Leistung nicht exakt getroffen, da auch hier alle Restriktionen erfüllt werden müssen und das HW zum Beispiel bereits mit seiner Minimalleistung einspeist.

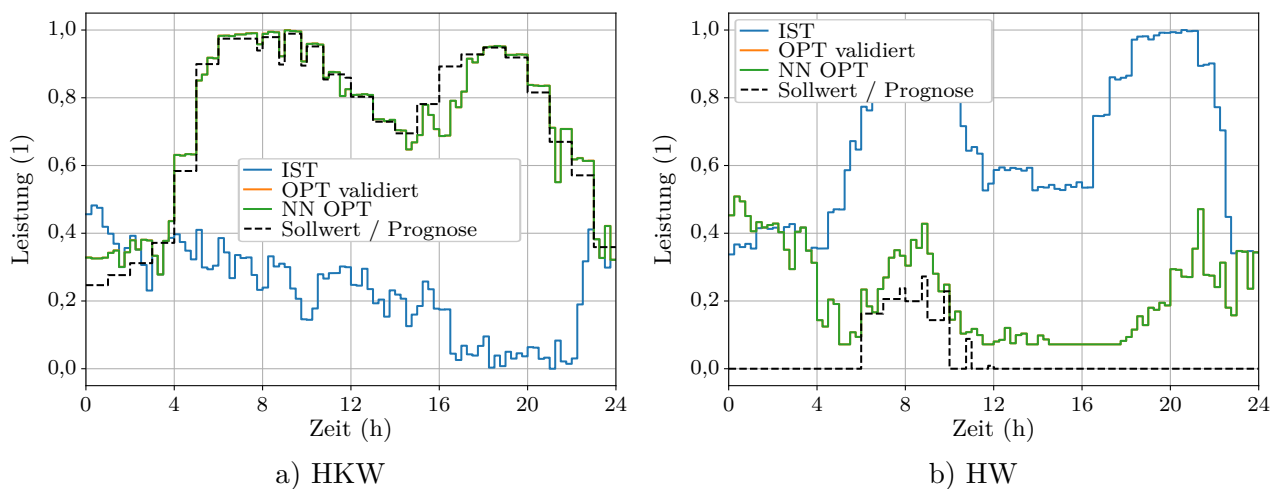


Abbildung 7.18.: Normierte Leistungen nach Optimierung von Fernwärmenetz 2 an Beispieltag 1 im Februar mit Leistungsvorgabe aus Prozessdaten und An-Vorgabe für das HW.

Zusammenfassend konnte in diesem Kapitel gezeigt werden, dass eine Optimierung der Fahrweise der beiden Wärmeerzeugungsanlagen in Fernwärmenetz 2 unter Vorgaben von Sollwerten und hydraulischen sowie thermischen Restriktionen auch bei großen Abweichungen der Prognosen von den real auftretenden Wärmeabnahmen möglich ist. Alle durchgeführten Optimierungen liefern innerhalb von weniger als einer Minute gute Ergebnisse. Mit dem Optimierer SLSQP ist in den meisten Fällen sogar eine Optimierung in unter einer Sekunde möglich.

7.2.3. Netz 2: Verwendung der Vorlaufleitungen als Temperaturspeicher

In Kapitel 3.3.1 wurde die Verwendung der Vorlaufleitungen als Temperaturspeicher beschrieben, um Lastspitzen abzufangen. Dabei wird die gesamte Einspeiseleistung ins Netz vorhergesagt und zu einigen Zeitpunkten eine höhere oder niedrigere Leistung durch das HKW in das Netz eingespeist. Der Massenstrom wird dabei jeweils von der Abnahme der Wärmetauscher im Netz bestimmt und die Leistung durch eine höhere Vorlauftemperatur erhöht bzw. durch eine niedrigere Temperatur reduziert. In den stationären Simulationen und Optimierungen ist eine Berücksichtigung dieses Effekts nicht möglich. Eine zusätzliche Beladung des Vorlaufspeichers wird nicht berücksichtigt. Bei Vorgabe des Beladens durch die wirtschaftliche Optimierung wird von der Optimierung also nur die Netzlast gedeckt, die Wärmeabnahmen werden korrekt simuliert. In der Praxis könnte die Vorlauftemperatur dennoch für das Einspeichern erhöht werden. Dies hätte lediglich höhere Temperaturen im Vorlauf zur Folge und keinen direkten Einfluss auf Massenströme oder Restriktionen wie Drücke und Strömungsgeschwindigkeit.

Beim Entladen des Speichers wird theoretisch Leistung aus dem Vorlaufspeicher entnommen. Diese wird in der Optimierung zusätzlich vom HKW eingespeist. Somit werden die gesamten Wärmeabnahmen wieder korrekt simuliert. Am HKW wird ebenfalls ein annähernd korrekter Massenstrom eingespeist, da die Netzumwälzpumpen die Differenzdrücke entsprechend der vorhandenen Abnahme einstellen müssen. Auch die Restriktionen werden somit korrekt berechnet. Im Realbetrieb könnte die Temperatur des HKW abgesenkt werden. So würde bei gleichem Massenstrom eine geringere Leistung eingespeist werden.

Insgesamt werden Massenströme und Restriktionen mit diesem Vorgehen sowohl bei der Be- als auch Entladung des Vorlaufspeichers möglichst genau berechnet. Abweichungen entstehen dennoch durch die inkorrekte Temperaturverteilung bei Verwendung einer höheren oder niedrigeren Vorlauftemperatur in der Realität. Hier müssen die Optimierungsergebnisse noch getestet werden. Alternativ kann eine Optimierung mit Verwendung instationärer Modelle untersucht werden. Dies hätte kompliziertere Modelle und längere Rechenzeiten zur Folge, könnte aber bessere Ergebnisse und die direkte Berechnung des Vorlaufspeichers ermöglichen.

7.2.4. Minimierung der Wärmeverluste in Fernwärmenetz 2

Zuletzt wird kurz eine weitere Optimierung mit dem Ziel der Minimierung der Wärmeverluste untersucht. Dabei werden die gleichen Modelle und Restriktionen wie bei den vorherigen Optimierungen und als Zielfunktion die folgende Gleichung (7.24) verwendet.

$$f = 0,5 \cdot \dot{Q}_{Verlust}^2 \quad (7.24)$$

Den Wärmeerzeugungsanlagen werden nur ihre Leistungsgrenzen vorgegeben und beide Anlagen sind permanent im An-Zustand. Die Gesamtleistung wird mit der gemessenen IST-Leistung von Beispieltag 1 festgelegt und die Temperaturgrenzen entsprechen denen der wirtschaftlichen Vortagesoptimierung. Die Optimierung funktioniert wie erwartet und reduziert die Vorlauftemperaturen auf die unteren Grenzkurven, wie in den Abbildungen 7.19a und 7.19b dargestellt. Dies führt zu einem reduzierten Wärmeverlust entsprechend Abbildung 7.19c. Über den gesamten Tag werden etwa 4,7 MW h eingespart.

Die hier nicht abgebildeten Wärmeerzeugungsleistungen weichen allerdings stark von den wirtschaftlich optimalen Sollwerten ab und weisen große Gradienten auf. Für diese sollte für eine sinnvolle Optimierung eine Begrenzung eingeführt werden. Zudem ist die starke Abweichung des HKW in der

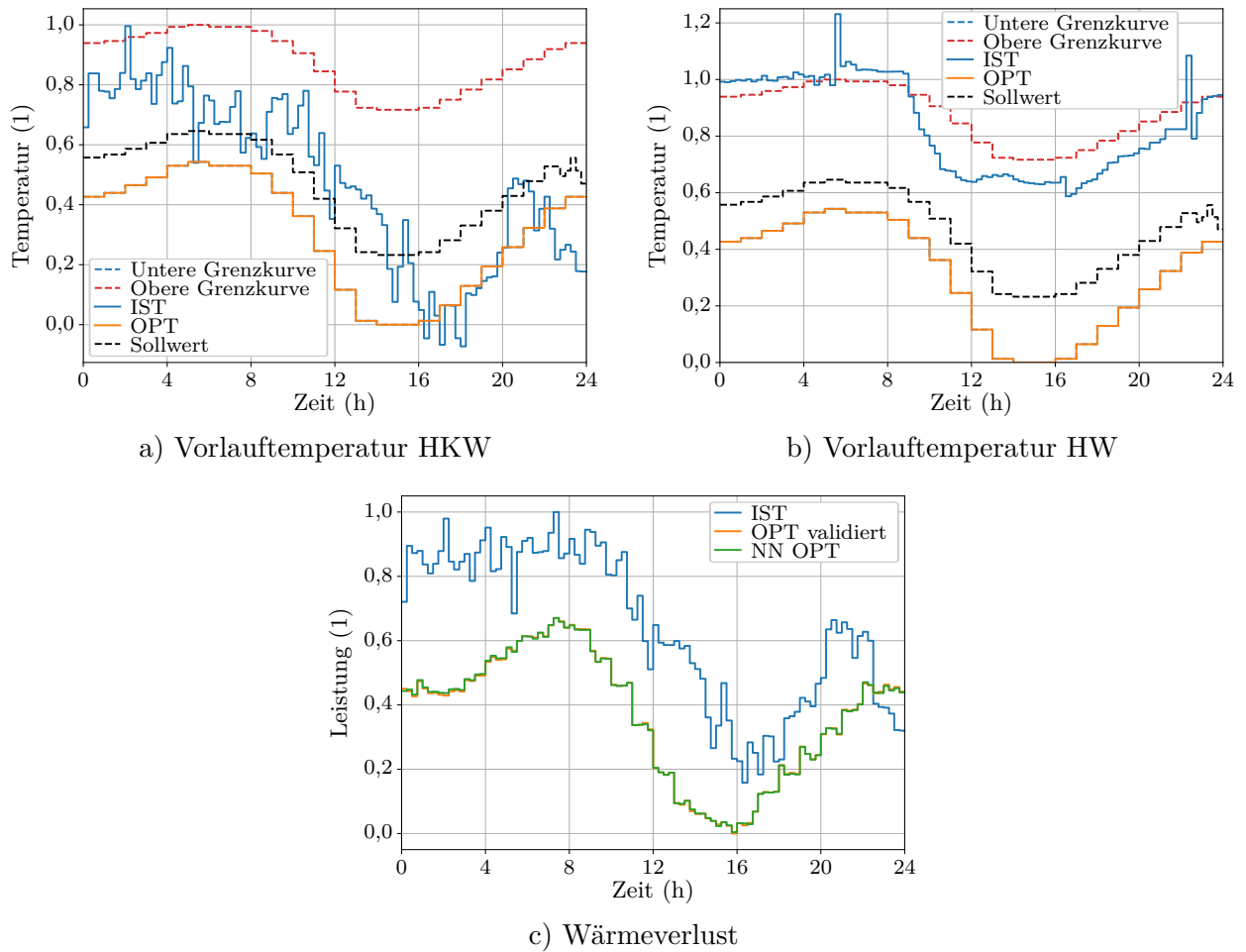


Abbildung 7.19.: Normierte Simulationsergebnisse nach Minimierung der Wärmeverluste von Fernwärmenetz 2 an Beispieltag 1 im Februar.

Praxis nicht akzeptabel, da dieses auch Strom produziert. Insgesamt kann mit der vorgestellten Optimierung aber auch eine Minimierung der Wärmeverluste durchgeführt werden. Ebenfalls wurde eine Minimierung der Summe der Wärmeverluste und Pumpenleistungen getestet und ist funktionsfähig. Die Optimierung kann also mit unterschiedlichen Zielfunktionen erfolgreich umgehen, muss für diese weiteren Anwendungen aber noch zum Beispiel um eine Gradientenbegrenzung der Leistungen oder Massenströme ergänzt werden.

7.3. Optimierung von Fernwärmenetz 3

Fernwärmenetz 3 ist mit vier Wärmeerzeugungsanlagen und einem großen, stark vermaschten Rohrleitungsnetz deutlich komplexer als die bereits untersuchten Netze. Die in den vorherigen Kapiteln erarbeiteten Optimierungsansätze können so weiter getestet und verbessert werden. Zudem ergeben sich durch eine Geothermieanlage und die DEA mit fünf Betriebsarten und der Möglichkeit der Drehzahlvorgabe zusätzliche Nebenbedingungen. Anhand dieses Netzes soll nun das zentrale Ziel der Arbeit, die Optimierung der verteilten regenerativen Wärmeeinspeisung, untersucht werden.

Die Optimierung erfolgt erneut nach dem in Abbildung 7.13 vorgestellten Schema. Allerdings werden hier jeweils vier Wärmeerzeugungsleistungen und Vorlauftemperaturen und zusätzlich die Druckhaltung des HKW sowie die Betriebsart der DEA optimiert. Von der wirtschaftlichen Optimierung wird zudem keine Gesamteinspeiseleistung vorgegeben, sondern eine Einspeiseleistung pro Teilgebiet. Diese Teilleistungen wurden auch als Eingangsdaten für die neuronalen Netze für quasi-stationäre Simulationen des *SIR 3S*[®] Modells verwendet, welche nun als Netzsimulation entsprechend Kapitel 6.3.2 zum Einsatz kommen. Somit gehen die Teilleistungen direkt als Eingang in die Optimierung. Die folgenden Nebenbedingungen (Restriktionen) sind von der Optimierung einzuhalten:

- Nenndrehzahl als maximale Drehzahl für alle Pumpen (n_{Pumpe})
- Minimale Drehzahl der DEA
- Minimale und maximale Massenströme (\dot{m}) und Leistungen (P) des HKW und der HW
- Maximale Strömungsgeschwindigkeit (v) an den Wärmeerzeugungsanlagen und der DEA
- Minimaler Differenzdruck (Δp) an den Schlechtpunkten (SP) der beiden Teilgebiete (TG)
- Minimale Vorlauftemperatur (T_{VL}) von 90 °C (an allen Kundenanlagen)
- Minimaler und maximaler Rücklaufdruck (p_{RL}) an den Schlechtpunkten der beiden Teilgebiete
- Maximaler Vorlaufdruck (p_{VL}) an den Schlechtpunkten der beiden Teilgebiete
- Maximale Netzeinspeisetemperaturgradienten (am Ausgang der Stationen)
- Außentemperaturabhängige Vorlauftemperaturgrenzen für die Wärmeerzeugungsanlagen entsprechend der wirtschaftlichen Optimierung für das HKW und HW 1 und 2
- Außentemperaturabhängige Vorlauftemperaturkurve für HW 3 nach dessen Betriebshandbuch. Von der Temperaturkurve wird eine Abweichung um ± 2 °C akzeptiert, um der Optimierung mehr Freiraum zu geben und eine leichte Variation der Temperatur zu ermöglichen.
- Maximaler Fehlmassenstrom von 10 t/h (siehe Kapitel 6.3.2)

Die verwendeten Restriktionen basieren in Teilen auf Angaben des Netzbetreibers und wurden um eigene Annahmen ergänzt, um eine vollständige Optimierung zu ermöglichen. Wie bei den vorherigen Netzen besteht jederzeit die Möglichkeit, die Restriktionen, das *SIR 3S*[®] Modell oder die neuronalen Netze anzupassen.

Anzumerken ist, dass in dem *SIR 3S*[®] Modell noch keine Regelung der Druckhaltung des HKW implementiert wurde. Diese sollte ursprünglich von der zu entwickelnden Optimierung übernommen werden. Allerdings werden nicht alle dafür notwendigen Restriktionen im *SIR 3S*[®] Modell bestimmt. So geht zum Beispiel der Mittendruck in HW 3 in die Schlechtpunktauswahl für den minimalen Rücklaufdruck in Teilgebiet 2 ein, wird aber in dem Netzmodell nicht berechnet. Hier wurde untersucht, den Mittendruck mit neuronalen Netzen basierend auf Messdaten vorherzusagen. Es konnte allerdings keine gute Übereinstimmung erreicht werden. Somit kann die Regelung der Druckhaltung nicht korrekt durchgeführt werden. Die Restriktionen, die dazu notwendig sind, sind bereits in der Optimierung

vorhanden, führen aber in dem derzeitigen Status zu keinen guten Ergebnissen. Die Druckhaltung wird anschließend in der Optimierung nur so eingestellt, dass zusammen mit der Betriebsart der DEA alle Restriktionen eingehalten werden. Dies entspricht in Folge nicht immer der Fahrweise, welche tatsächlich umgesetzt werden würde und Unterschiede zwischen Druckvorhersage und Realität können entstehen. Dennoch wäre dies eine mögliche Fahrweise, welche alle Restriktionen erfüllt. Vor einem realen Einsatz der Optimierung müssen die restlichen Restriktionen von dem *SIR 3S*[®] Modell ausgegeben werden. Anschließend kann die Druckhaltung korrekt implementiert und von der Optimierung eingestellt werden. Alternativ wäre es vorteilhaft, die Regelung der Druckhaltung direkt in dem *SIR 3S*[®] Modell zu implementieren. So könnte das neuronale Netz die Regelung erlernen und den Aufwand der Optimierung reduzieren. Rechenzeiten können dann weiter reduziert werden.

Im Folgenden findet zuerst eine Untersuchung der DEA-Fahrweise statt. Die unterschiedlichen Betriebsarten der DEA führen zusammen mit nicht-linearen Restriktionen zu einem gemischt-ganzzahligen nicht-linearen Optimierungsproblem, welches möglichst umgangen werden sollte. Zudem sind Betriebsartenwechsel häufig unerwünscht. In Kapitel 7.3.1 wird deshalb versucht, durch eine veränderte Lastverteilung im Netz auch bei konstant vorgegebener Betriebsart eine mögliche Fahrweise zu finden. Anschließend wird in Kapitel 7.3.2 die Optimierung inklusive eines Vorgehens für die DEA und die Geothermieanlage in HW 3 vorgestellt. Dabei werden wieder Vergleiche mehrerer Optimierer durchgeführt, der Einfluss der Rücklaufftemperaturvorgabe untersucht sowie Ergebnisse mit instationären Simulationen validiert. Analog zu Netz 2 werden in den Kapiteln 7.3.3 bis 7.3.5 das Vorgehen bei abweichenden Wärmelastprognosen, die Vorlaufftemperaturspeicherung und die Minimierung der Wärmeverluste untersucht.

7.3.1. Fahrweise der Druckerhöhungsanlage

Betriebsartenwechsel der DEA führen während einer Optimierung zu Sprüngen in den nicht-linearen Zielfunktionen und Nebenbedingungen. Zudem sind mehrere der an den Beispieltagen aufgenommenen Betriebsartenwechsel auch aus Sicht des Netzbetriebes unerwünscht. Einige der Wechsel konnten bereits durch Analysen der DEA und Implementierung baulicher und regelungstechnischer Maßnahmen behoben werden. Andere Wechsel werden dadurch verursacht, dass es zu Wechseln der Wärmetransportrichtung zwischen den Teilgebieten 1 und 2 kommt. Hier wird eine Untersuchung der Betriebsartenwechsel an Beispieltag 1 durchgeführt. Die Betriebsarten sind in Abbildung 7.20 dargestellt und erläutert. Fast den gesamten Morgen befindet sich die DEA in der Betriebsart, in welcher Wärme von Teilgebiet 1 mittels Vorlaufpumpen in Teilgebiet 2 transportiert wird. Um 4 Uhr werden die Pumpen kurz ausgeschaltet. Mittags wechselt die Transportrichtung für einige Minuten und Wärme wird durch die Rücklaufpumpen gefördert. Anschließend ist nachmittags wieder überschüssige Wärme in Teilgebiet 1 vorhanden, welche auch ohne die Pumpen der DEA in Teilgebiet 2 gelangt. Um 23:15 Uhr findet erneut ein Richtungswechsel statt.

Es soll untersucht werden, wie eine Änderung der Einspeiseleistungen des HKW und der HWs eine konstante Fahrweise der DEA ermöglichen kann. So wird zunächst eine Simulation mit den gemessenen Eingangswerten von Beispieltag 1 und der DEA in einer konstanten Betriebsart ohne Pumpeneinsatz durchgeführt. Abbildung 7.21a zeigt die resultierenden Massenströme in Teilgebiet 2 und der DEA. An den vier markierten Zeitpunkten ist der Massenstrom der DEA negativ (normiert unter 0) und die Transportrichtung wechselt somit. Die Zeitpunkte sind (bis auf 8 Uhr) auch jene, zu welchen die Betriebsart der DEA an Tag 1 wechselte.

Für eine konstante Fahrweise der DEA muss zu diesen Zeitpunkten ein Wärmetransport von Teilgebiet 1 in Teilgebiet 2 stattfinden. Dies ist durch eine Anpassung der eingespeisten Leistung im HW 3 in Teilgebiet 2 möglich. Als Beispiel wird der Massenstrom des HW zu den vier Zeitpunkten um etwa 5% reduziert. Als Konsequenz erhöht das HKW in Teilgebiet 1, welches den Differenzdruck im

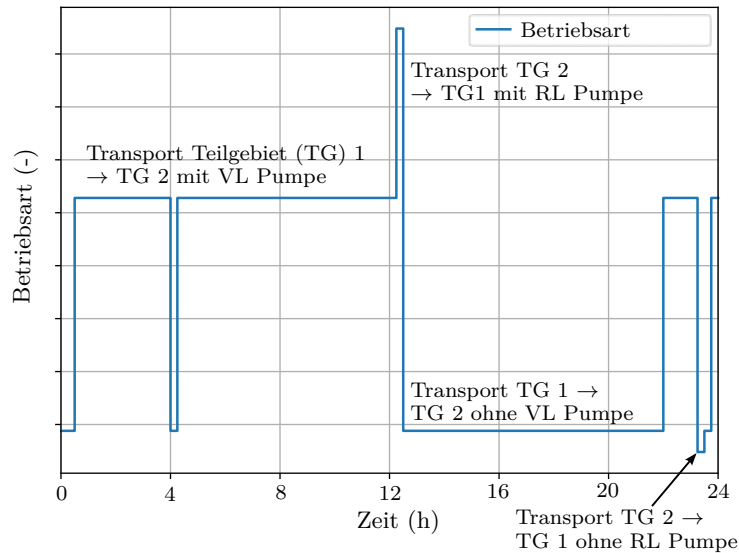
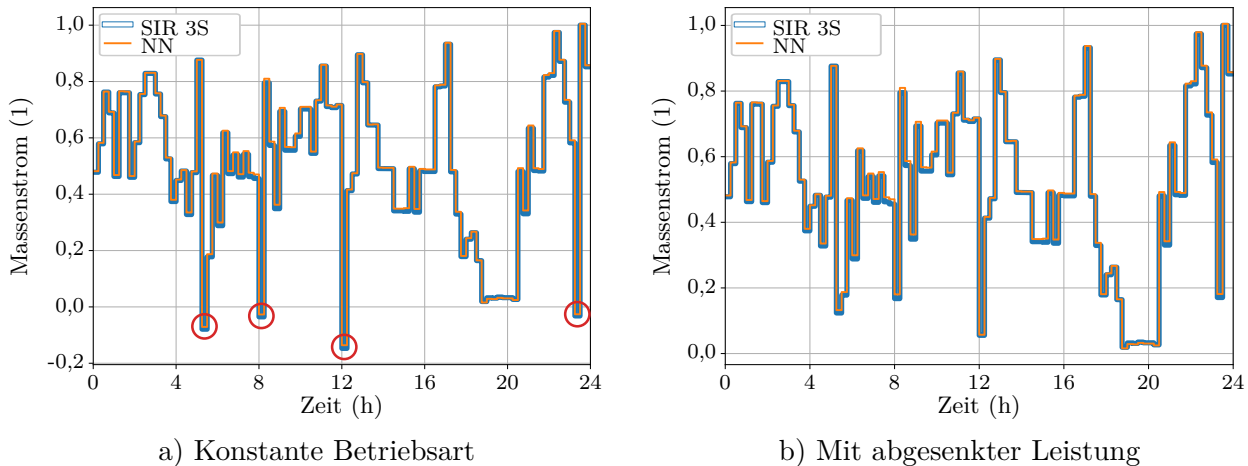


Abbildung 7.20.: Betriebsarten der DEA in Fernwärmenetz 3 an Beispieltag 1 (Februar).



a) Konstante Betriebsart

b) Mit abgesenkter Leistung

Abbildung 7.21.: Normierte Massenströme der DEA an Beispieltag 1 mit konstanter Betriebsart und mit konstanter Betriebsart sowie abgesenkter Leistung des HW 3 in Teilgebiet 2.

Netz regelt, seine Leistung um die am HW reduzierte Leistung. Das Ergebnis der Rechnung mit reduzierten Massenströmen, der DEA in konstanter Betriebsart und den restlichen Eingangsdaten von Beispieltag 1 ist in Abbildung 7.21b zu sehen. Die Transportrichtung ist nun zu allen Zeitpunkten konstant (normierter Massenstrom über 0) und die Pumpen der DEA müssen nicht eingesetzt werden. Eine geringe Anpassung der Fahrweise des HW resultiert also schon in einer Verbesserung für die DEA. Diese wird mit in die zu entwickelnde Optimierung einbezogen, um so eine Fahrweise mit möglichst wenig Umschaltvorgängen zu erreichen.

7.3.2. Optimierung von Netz 3 nach Leistungsprognose und wirtschaftlichen Sollwerten

Nun wird eine 24-Stunden-Optimierung von Beispieltag 1 mit Sollwerten der wirtschaftlichen Optimierung sowie deren Leistungsprognose getestet. Verwendet wird das neuronale Netz für stationäre Simulationen aus Kapitel 6.3.2 mit der Rücklaufemperatur als Eingang. Diese wird dabei fix entsprechend den Messwerten an Beispieltag 1 vorgegeben. Als Optimierungsalgorithmen kommen wieder der lokale Ansatz SLSQP und der globale Ansatz *differential evolution* zum Einsatz. Für die Optimierung stehen acht Variablen zur Verfügung. Diese sind die Leistungen der drei HWs und ihre Vorlauftemperaturen sowie die des HKW. Die Leistung des HKW ergibt sich aus der Leistungsbilanz

mit den Heizwerken und der gesamt einzuspeisenden Leistung, welche entsprechend der Leistungsprognose fix ist. Zudem kann die Druckhaltung variiert werden, welche wie bereits beschrieben nicht der realen Regelung entspricht. Bei den Leistungen der HWs wird das *Unit-Commitment*-Problem als von der wirtschaftlichen Optimierung gelöst betrachtet. Bei einer Sollleistung von 0 wird ein HW in der Optimierung somit nicht berücksichtigt.

Um eine gemischt ganzzahlige nicht-lineare Optimierung zu vermeiden, wird vor Beginn einer Optimierung für jeden Zeitschritt die Flussrichtung über die DEA berechnet. Dafür wird zunächst geprüft, ob die Leistung von HW 3 kleiner ist als die prognostizierte Einspeiseleistung in Teilgebiet 2. Ist dies der Fall, fließt über die DEA ein Massenstrom von Teilgebiet 1 in Teilgebiet 2. Bei größerer Leistung des HW 3 fließt der Massenstrom entsprechend umgekehrt.

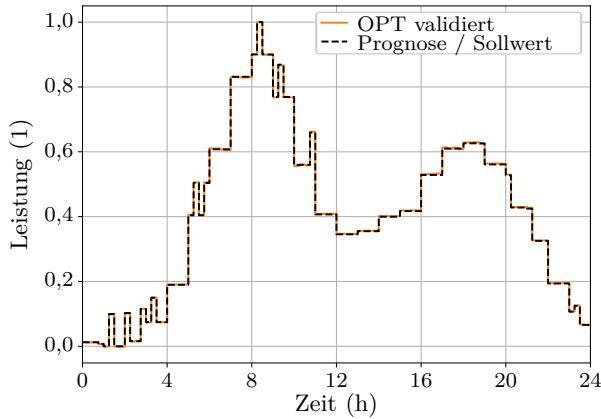
Durch die Geothermieanlage in HW 3 ergibt sich eine weitere Problematik. Von der wirtschaftlichen Optimierung werden an Wintertagen teilweise hohe Leistungen der Geothermie und niedrige Leistungen der Gasturbinen vorhergesagt. Dieser Zustand ist nicht immer möglich, da bestimmte Vorlauftemperaturgrenzen gelten und diese so nicht eingehalten werden. Die Geothermie speist mit Temperaturen unter 90 °C ein, welche an Wintertagen zu niedrig sind. Hier wird vor der Optimierung die nötige Gasleistung berechnet, um die minimal vorgegebene Vorlauftemperatur zu erreichen. Diese Leistung zusammen mit der Geothermieleistung wird dann für die Bestimmung der Flussrichtung verwendet. Der DEA wird anschließend für die Optimierung jeweils die Betriebsart mit der berechneten Flussrichtung ohne Pumpe vorgegeben. So kann für viele Fälle bereits ohne weitere Optimierung eine mögliche Betriebsart der DEA definiert werden. Ist die Optimierung ohne Pumpe nicht erfolgreich, wird erneut mit Pumpeneinsatz optimiert. Ist dies auch nicht erfolgreich, wird die Optimierung mit umgekehrter Flussrichtung versucht, jeweils ohne und mit Pumpe. Bei Pumpeneinsatz wird zudem die Drehzahl der DEA abgefragt. Bei Unterschreiten des Minimums werden die Drehzahlen der Pumpen auf das Minimum festgelegt. Mit diesem Vorgehen kann für alle Beispieltage eine Fahrweise ohne Pumpeneinsatz erreicht werden. Die Flussrichtung und somit die Betriebsart der DEA wechseln zwar potenziell, allerdings kommt es ohne Pumpeneinsatz nicht zu Sprüngen in den minimalen Drücken oder Differenzdrücken.

Als Zielfunktion wird die folgende Gleichung (7.25) verwendet, welche alle Abweichungen der Leistungen und Vorlauftemperaturen von den wirtschaftlich optimierten Sollwerten mit einbezieht. Die Einzelkomponenten der Zielfunktion sind mit Gewichtungsfaktoren versehen. Das HKW erhält mit 100 die höchste Gewichtung, da dieses auch für die Stromproduktion zum Einsatz kommt. Verschiedene Gewichtungen wurden getestet und bei Bedarf ist eine Änderung möglich, um andere Optimierungsziele zu erreichen.

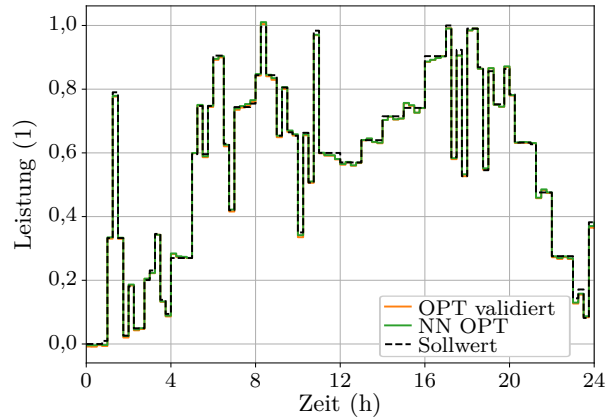
$$\begin{aligned}
 \Delta P_{HKW} &= 100 \cdot (P_{HKW} - SW_{P_{HKW}}) \\
 \Delta P_{HW1} &= 10 \cdot (P_{HW1} - SW_{P_{HW1}}) \\
 \Delta P_{HW2} &= 1 \cdot (P_{HW2} - SW_{P_{HW2}}) \\
 \Delta P_{HW3} &= 10 \cdot (P_{HW3} - SW_{P_{HW3}}) \\
 \Delta T_{VL, HKW} &= 100 \cdot (T_{VL, HKW} - SW_{T_{VL, HKW}}) \\
 \Delta T_{VL, HW1} &= 1 \cdot (T_{VL, HW1} - SW_{T_{VL, HW1}}) \\
 \Delta T_{VL, HW2} &= 1 \cdot (T_{VL, HW2} - SW_{T_{VL, HW2}}) \\
 \Delta T_{VL, HW3} &= 1 \cdot (T_{VL, HW3} - SW_{T_{VL, HW3}}) \\
 f &= 0,5 \cdot (\Delta P_{HKW}^2 + \Delta P_{HW1}^2 + \Delta P_{HW2}^2 + \Delta P_{HW3}^2 \\
 &\quad + \Delta T_{VL, HKW}^2 + \Delta T_{VL, HW1}^2 + \Delta T_{VL, HW2}^2 + \Delta T_{VL, HW3}^2) \quad (7.25)
 \end{aligned}$$

Die Abbildungen 7.22a und 7.22b zeigen die Optimierungsergebnisse für die gesamte Einspeiseleistung und die des HKW. Beide Leistungen folgen dem Sollwert. Stärker weichen die Leistungen von HW 1

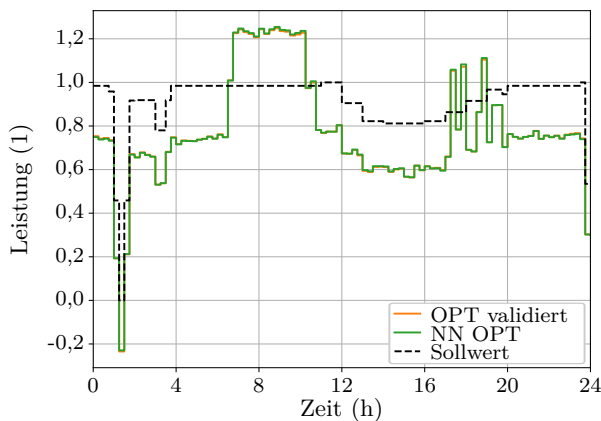
und 3 in Teilgebiet 1 und 2 von den Sollwerten ab, wie in den Abbildungen 7.22c und 7.22d dargestellt. HW 3 speist häufig eine höhere Leistung als vorgegeben ein, da der Sollwert der Geothermieleistung um eine Leistung der Gasturbinen zum Erwärmen der Vorlauftemperatur ergänzt werden muss. So wird den gesamten Tag über ein Massenstrom von Teilgebiet 2 in Richtung Teilgebiet 1 gefördert. Auch morgens, etwa zwischen 7 und 10 Uhr, ist eine Abweichung der Leistungen sichtbar. Während die Leistung von HW 3 nicht bis zum Sollwert erhöht wird, wird die Leistung von HW 1 über dessen Sollwert gesteigert. Die Optimierung gibt dies vor, um alle Restriktionen einzuhalten.



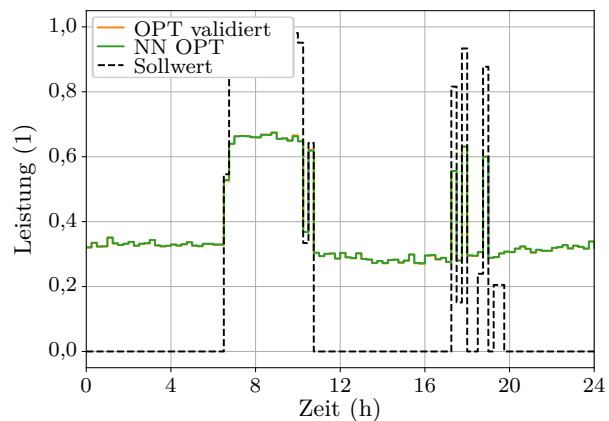
a) Gesamtleistung



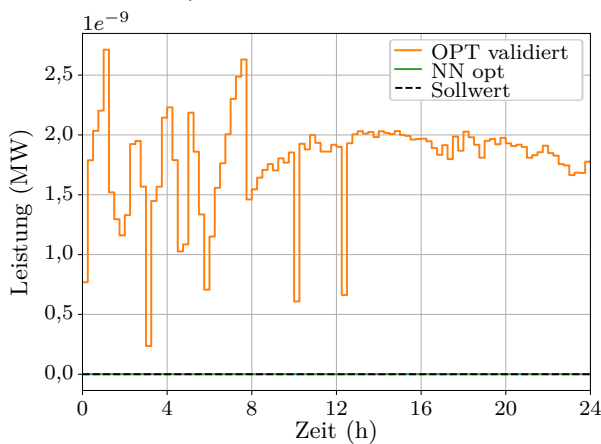
b) Leistung des HKW



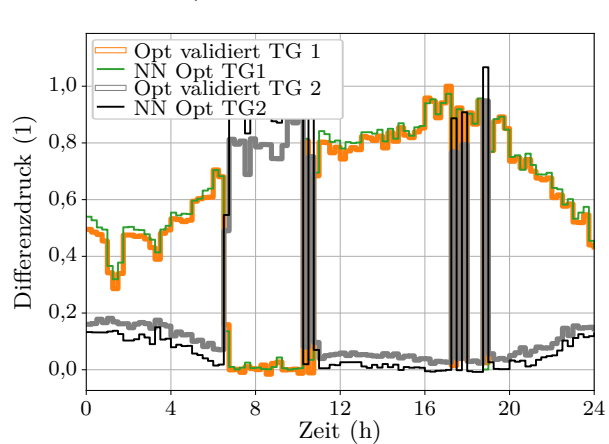
c) Leistung von HW 1



d) Leistung von HW 3



e) Leistung von HW 2



f) Minimaler Differenzdruck

Abbildung 7.22.: Normierte Simulationsergebnisse für Beispieltag 1 im Februar nach stationärer Optimierung von Fernwärmenetz 3.

Abbildung 7.22e zeigt die Einspeiseleistung von HW 2. Wie vorgegeben wird das Heizwerk nicht für die Optimierung verwendet und befindet sich im Aus-Zustand. In der Validierungsrechnung von *SIR 3S*[®] entsteht dennoch eine kleine, zu vernachlässigende Leistung. In Abbildung 7.22f sind zudem die minimalen Differenzdrücke an den Schlechtpunkten in beiden Teilgebieten abgebildet. Der Grenzwert liegt in der normierten Darstellung bei 0,0 für beide Teilgebiete. Beide Grenzwerte können von der Optimierung eingehalten werden. Für Teilgebiet 2 sind einige Abweichungen zwischen dem neuronalen Netz (NN) und der *SIR 3S*[®] Simulation ("Opt validiert") zu erkennen. Diese wurden bereits in Kapitel 6.3.2 beobachtet. An dieser Stelle können die neuronalen Netze noch nachgebessert werden. Interessant ist noch die Verschiebung des Differenzdrucks bei Betrieb von HW 3. Erkennbar ist, dass sich der Netzschlechtpunkt während der Leistungspitzen von HW 3 von Teilgebiet 2 zu Teilgebiet 1 verschiebt. Die Leistung von HW 3 kann also nicht ganz bis zum Sollwert erhöht werden, da sonst in Teilgebiet 1 der Differenzdruck unter den Minimalwert fällt. Die Optimierung liefert plausible Ergebnisse.

Die optimierten Vorlauftemperaturen des HKW und von HW 1 sind in den Abbildungen 7.23a und 7.23b dargestellt. Der Vorlauftemperatursollwert des HKW wird entsprechend der hohen Gewichtung in der Zielfunktion sehr gut eingehalten. Für HW 1 weicht die Temperatur stärker ab. Allerdings ist das Temperaturband zwischen unterer und oberer Grenzkurve nur etwa 1 °C groß. Somit schwankt die Temperatur auch hier nicht stark. Für HW 3 bewegt sich die nicht dargestellte Vorlauftemperatur ebenso zwischen der unteren und oberen Grenze.

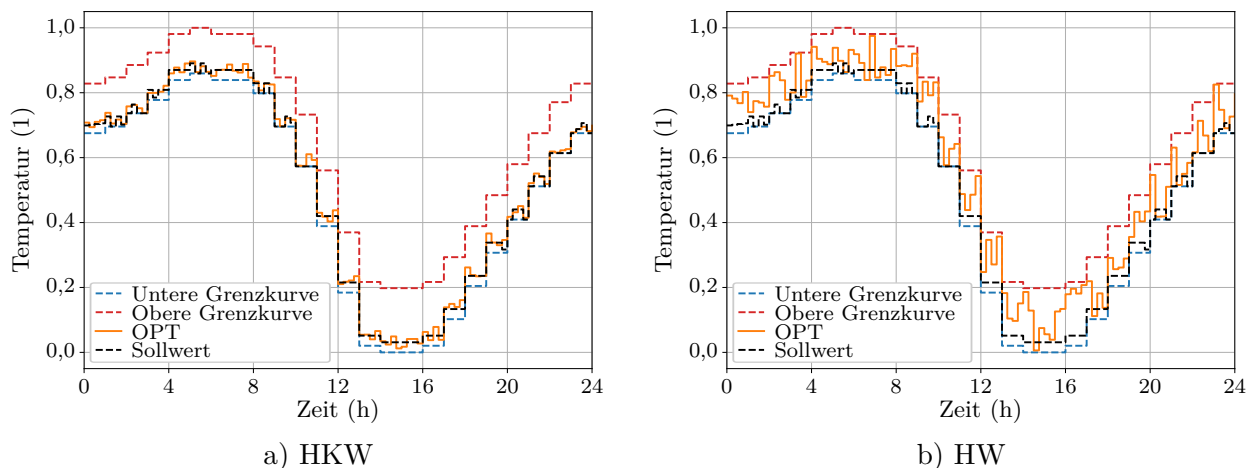


Abbildung 7.23.: Normierte Vorlauftemperaturen nach stationärer Optimierung von Fernwärmenetz 3 an Beispieltag 1 im Februar.

Mit dem neuronalen Netz ohne Rücklauftemperatur ist eine Optimierung ebenfalls möglich. Die Abweichungen zwischen den Optimierungsergebnissen des neuronalen Netzes und der Validierungssimulation in *SIR 3S*[®] sind so etwas höher. Dennoch können alle Restriktionen und die Sollwerte für Leistungen und Vorlauftemperaturen gut eingehalten werden. Eine Rücklauftemperaturvorhersage wurde für Netz 3 bisher nicht umgesetzt. Auch hier kann aber eine Vorhersage mittels neuronaler Netze basierend auf Messdaten nachträglich implementiert werden. Für die zwei weiteren Beispieltage ist die Optimierung funktionsfähig und liefert gute Ergebnisse innerhalb der Grenzwerte. Tabelle 7.3 zeigt die Dauer der Optimierungen für die Beispieltage mit den neuronalen Netzen mit und ohne Rücklauftemperatur.

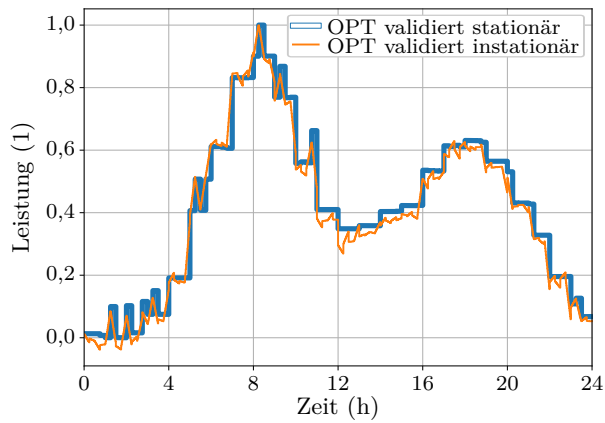
Tabelle 7.3.: Optimierungsdauer für Netz 3 an den drei Beispieltagen mit Leistungsvorgabe aus der Vortags-Prognose.

	Dauer mit <i>differential evolution</i> (s)
Tag 1 (Februar) mit T_{RL} Vorgabe	14,23
Tag 1 (Februar) ohne T_{RL} Vorgabe	16,19
Tag 2 (Mai) mit T_{RL} Vorgabe	27,99
Tag 2 (Mai) ohne T_{RL} Vorgabe	31,14
Tag 3 (Juli) mit T_{RL} Vorgabe	30,61
Tag 3 (Juli) ohne T_{RL} Vorgabe	30,30

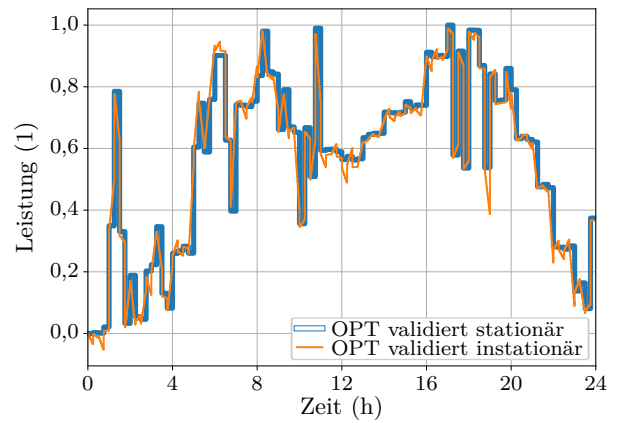
Die Optimierung mit dem lokalen Ansatz SLSQP scheitert in allen Optimierungsfällen. Die gradientenbasierte Optimierung findet entweder keine Lösung innerhalb der maximal angesetzten 1000 Iterationen oder kann keine geeignete Suchrichtung festlegen, in welcher die Zielfunktion abnimmt. Dieses Ergebnis war für lokale Optimierungen bei zunehmender Anzahl an Variablen zu erwarten. Theoretisch könnte noch die Startwertabhängigkeit der Optimierung untersucht werden. Bisher wurden die Sollwerte als Startwerte verwendet. Andererseits liefert der globale Ansatz *differential evolution* sehr gute Ergebnisse. Die Optimierungen mit und ohne Rücklauftemperaturvorgabe finden in den drei Fällen in unter 32s ein Ergebnis. Für den Einsatz in einer durch Personal zu validierenden Optimierung sind diese Zeiten schnell genug. Auch für den Einsatz in einer MPC sind die Optimierungszeiten nach derzeitiger Einschätzung für träge thermo-hydraulische Systeme wie Fernwärmenetz ausreichend. Für einen weitere Beschleunigung der Optimierungen kann der lokale Ansatz im Detail untersucht werden.

Wie bei Fernwärmenetz 2 ist eine Überprüfung der Optimierungsergebnisse mit einer instationären Simulation in *SIR 3S*[®] notwendig. Die optimierten Eingangswerte der stationären *SIR 3S*[®] Simulation (OPT validiert) werden daher für eine instationäre 24-Stunden-Simulation mit 1-Minute-Zeitschritten verwendet. Die Abbildungen 7.24a und 7.24b zeigen die gesamte Einspeiseleistung und die des HKW. Die stationäre Simulation stimmt in beiden Fällen gut mit der instationären Simulation überein. Zu einigen Zeitpunkten wird in der instationären Simulation eine insgesamt zu geringe Leistung eingespeist. Die maximale Abweichung beläuft sich auf etwa 2,7%. Abbildung 7.24c zeigt zudem den minimalen Differenzdruck in Teilgebiet 1 und Abbildung 7.24d die maximale Strömungsgeschwindigkeit des HKW. Während die instationär simulierte Strömungsgeschwindigkeit ähnlich der Leistung des HKW gut mit der stationären Simulation übereinstimmt, zeigt der minimale Differenzdruck deutliche Schwingungen und Druckspitzen. Diese entstehen bei Leistungsänderungen von HW 3, siehe Abbildung 7.22d. Die neuen Sollwerte werden in der instationären Simulation sprunghaft auf das Netzmodell gegeben und verursachen so Schwingungen. In der Realität ist eine PI(D)-Regelung auf der Feldebene vorhanden, welche die Leistung des Kraftwerks kontinuierlich anhebt oder absenkt. So werden die Schwingungen nicht in dem hier dargestellten Maße auftreten. Dennoch gibt es auch für den minimalen Differenzdruck im eingeschwungenen Zustand Abweichungen von etwa 5% zwischen instationärer und stationärer Simulation. Die Abweichungen der Leistungen, Differenzdrücke, etc. müssen im Einsatz von den Regelungen auf der Feldebene und gegebenenfalls dem Betriebspersonal kompensiert werden. Dieses Vorgehen muss noch getestet werden.

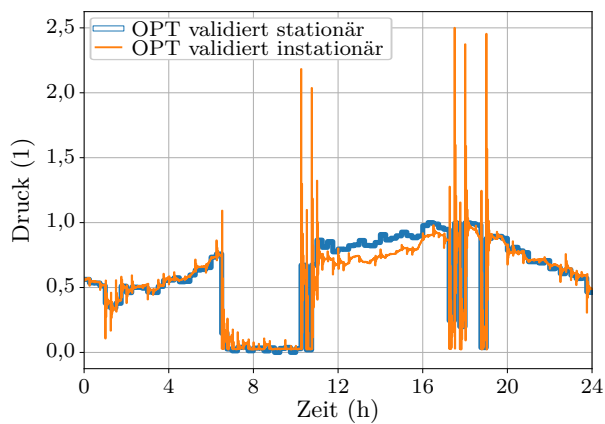
Insgesamt wurde in diesem Kapitel erfolgreich die verteilte Wärmeversorgung in Fernwärmenetz 3 mit vier Wärmeerzeugungsanlagen, einer komplexen Betriebsweise der DEA und einer Geothermieanlage in HW 3 optimiert. Drei Beispieltage waren für Tests verfügbar und können in unter 32s optimiert werden. Dabei zeigte sich, dass für Fernwärmenetzen mit einer hohen Anzahl an Wärmeerzeugungsanlagen und somit vielen Variablen globale Optimierungsansätze für das Finden einer optimalen Lösung notwendig sind.



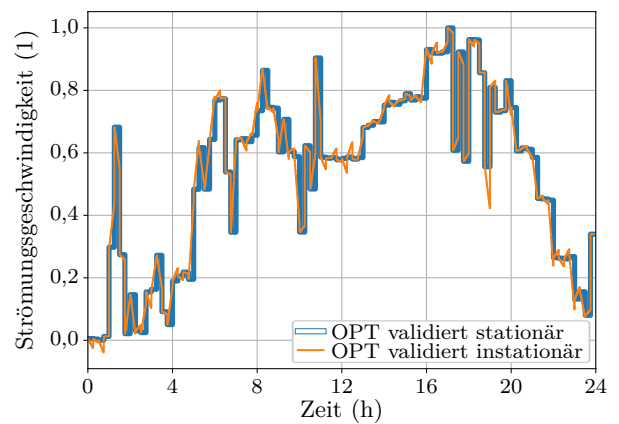
a) Gesamtleistung



b) Leistung des HKW



c) Minimaler Differenzdruck



d) Strömungsgeschwindigkeit am HKW

Abbildung 7.24.: Normierte stationäre und instationäre Simulationsergebnisse nach Optimierung von Fernwärmenetz 3 an Beispieltag 1 im Februar.

7.3.3. Optimierung von Netz 3 mit Einspeiseleistung aus Prozessdaten und wirtschaftlichen Sollwerten

Auch für die Optimierung von Fernwärmenetz 3 soll die Funktionsfähigkeit bei abweichenden Prognosen der gesamten Einspeiseleistung getestet werden. So werden die gleichen Sollwerte und Restriktionen wie in der vorherigen Optimierung verwendet, während die Einspeiseleistung entsprechend den Prozessdaten und nicht der Prognose vorgegeben wird. Abbildung 7.25a zeigt die am Vortag prognostizierte Einspeiseleistung sowie die tatsächlich gemessene IST-Leistung, welche etwas unter der Prognose liegt und die von der Optimierung erreichten Werte. Die Optimierung trifft die IST-Leistung wie gefordert sehr genau. In Abbildung 7.25b ist zudem die Leistung des HKW zu sehen. In der IST-Fahrweise wurde die Leistung des HKW deutlich über den Sollwerten gefahren, um die Differenz zwischen der Prognose und der tatsächlichen Leistungsabnahme auszugleichen. Nach der Optimierung kann der Sollwert gut erreicht werden.

In den Abbildungen 7.25c und 7.25d sind die Leistungen von HW 1 und 3 dargestellt. Im Vergleich zur IST-Fahrweise liegen diese näher an ihren Sollwerten. Für HW 3 ist wieder zu beobachten, dass dieses den Großteil der Zeit eine höhere Leistung als gefordert einspeist. Hier muss erneut die Gasturbine eingesetzt werden, um die Vorlauftemperatur zu erhöhen. Erkennbar ist zudem, dass die IST- und optimierte Fahrweise für HW 3 außerhalb der Einspeisespitzen gut übereinstimmen. Auch in der Realität wurde also eine Mindestleistung eingespeist, welche die Vorlauftemperaturanforderung erfüllt. HW 2 bleibt wie gefordert im Aus-Zustand.

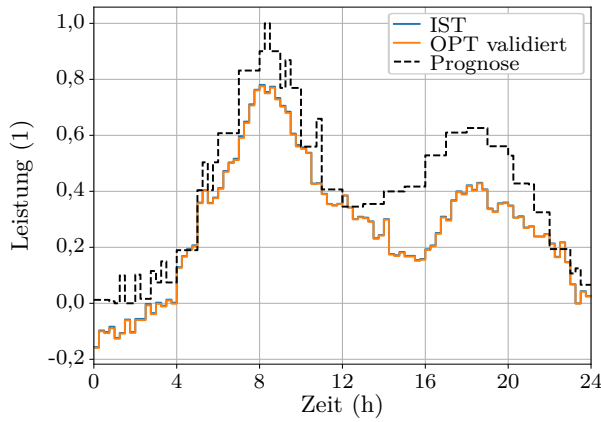
Die Soll-Vorlauftemperatur des HKW wird wie in der vorherigen Optimierung gut eingehalten und ist in Abbildung 7.25e zu sehen. Die IST-Vorlauftemperatur des Beispieltages zeigt wie bei Netz 2 teilweise Vorlauftemperaturen unter oder über den Grenzkurven. Der Grund sind die Grenzkurven, welche am Vortag berechnet werden und sich während eines Tagesverlaufs ändern können. Für die Optimierung in dieser Arbeit sind nur die Grenzkurven der Vortagesoptimierung bekannt, welche somit als Grenzwerte verwendet werden.

In Abbildung 7.25f ist zudem wieder der minimale Differenzdruck an den Schlechtpunkten in Teilnetz 1 und 2 abgebildet. Hier ist zu erkennen, dass der Differenzdruck in Teilnetz 1 bei Erhöhung der Leistung in HW 3 nicht so stark abfällt wie zuvor. Ursache ist die verglichen mit der Prognose geringere gesamte Einspeiseleistung. Folglich kann in diesem Fall die Leistung von HW 3 höher eingestellt werden und trifft genauer die Spitzen der Sollleistung (siehe Abbildung 7.25d).

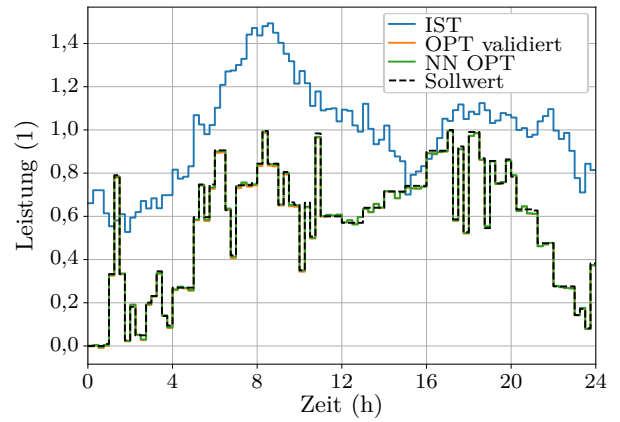
Die Optimierung mit von der Prognose abweichender Einspeiseleistung ist sowohl mit als auch ohne Rücklauftemperaturvorgabe für alle Beispieltage durch den Ansatz *differential evolution* funktionsfähig und liefert gute Ergebnisse. Die lokale Optimierung SLSQP schlägt dagegen in allen Fällen fehl. Die Rechenzeiten der verschiedenen Fälle sind in Tabelle 7.4 dokumentiert. Auch für diese Fälle ist eine Optimierung in unter einer Minute möglich.

Tabelle 7.4.: Optimierungsdauer für Netz 3 an den drei Beispieltagen mit Leistungsvorgabe aus Prozessdaten.

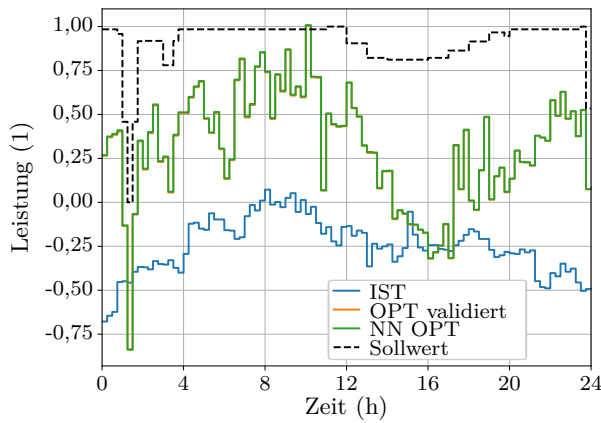
	Dauer mit <i>differential evolution</i> (s)
Tag 1 (Februar) mit T_{RL} Vorgabe	14,49
Tag 1 (Februar) ohne T_{RL} Vorgabe	14,42
Tag 2 (Mai) mit T_{RL} Vorgabe	19,15
Tag 2 (Mai) ohne T_{RL} Vorgabe	18,66
Tag 3 (Juli) mit T_{RL} Vorgabe	25,48
Tag 3 (Juli) ohne T_{RL} Vorgabe	26,71



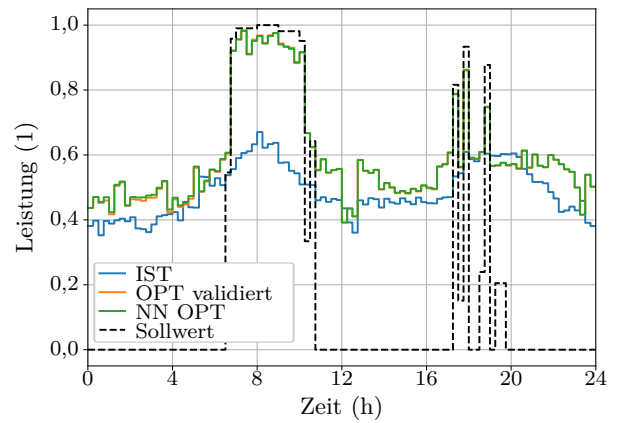
a) Gesamtleistung



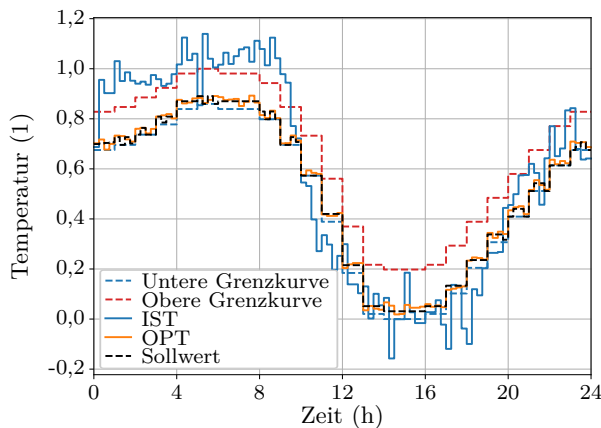
b) Leistung des HKW



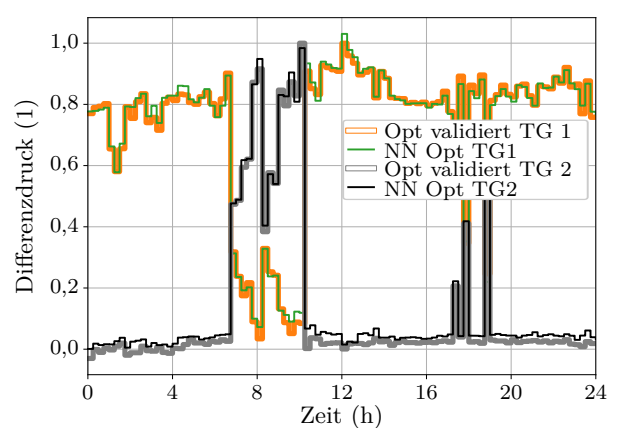
c) Leistung von HW 1



d) Leistung von HW 3



e) Vorlauftemperatur HKW



f) Minimaler Differenzdruck

Abbildung 7.25.: Normierte Simulationsergebnisse nach Optimierung von Fernwärmenetz 3 an Beispieltag 1 im Februar mit Leistungsvorgabe aus Prozessdaten.

7.3.4. Netz 3: Verwendung der Vorlaufleitungen als Temperaturspeicher

Für Fernwärmenetz 3 werden wie bei Fernwärmenetz 2 von der wirtschaftlichen Optimierung höhere oder niedrigere Vorlauftemperaturen vorgeschlagen, um das Vorlaufnetz als Temperaturspeicher zu verwenden. Dabei werden ebenfalls höhere oder niedrigere Wärmeeinspeiseleistungen berechnet. Mit stationären Simulationen kann dieses Vorgehen nicht abgebildet werden. Die Optimierung verwendet so nur die Leistung als Ziel, welche notwendig ist, um die Netzlast zu decken. Das gleiche Vorgehen wurde bereits für Fernwärmenetz 2 verwendet. Höhere oder niedrigere Temperaturen und somit Leistungen können anschließend dennoch vom Personal vorgegeben werden und sollten die hydraulischen und thermischen Restriktionen nicht negativ beeinflussen. Dies muss im Betrieb überprüft werden. Reicht diese Optimierung zusammen mit dem Einsatz vom Personal nicht aus, muss eine instationäre Simulation verwendet werden, welche in der Lage ist, das Speicherverhalten zu simulieren.

7.3.5. Minimierung der Wärmeverluste in Fernwärmenetz 3

Auch für Fernwärmenetz 3 wird eine Optimierung der Wärmeverluste untersucht. Die Restriktionen bleiben dabei wie in den vorherigen Optimierungen. Als Zielfunktion wird der halbe quadrierte Wärmeverlust verwendet. Wie für Netz 1 und 2 ist eine Optimierung prinzipiell möglich und resultiert in Vorlauftemperaturen der Wärmeerzeugungsanlagen, welche sich an den unteren Grenzkurven orientieren. An Beispieltag 1 im Februar können so etwa 7,5 MWh gespart werden. Allerdings resultiert dies wie in Netz 2 in einer Fahrweise, welche vor allem für das HKW stark von den Sollwerten abweichende Leistungen erfordert. Auch die Gradienten der Einspeiseleistungen des HKW und von HW 1 sind hoch. Eine Optimierung mit dem Ziel, sowohl Wärmeverluste als auch Pumpenleistungen zu reduzieren, zeigt ähnliche Ergebnisse. Insgesamt kann so gezeigt werden, dass verschiedene Zielfunktionen zum Einsatz kommen können. Bisher ist der Fokus auf die Einhaltung gegebener Sollleistungen und Vorlauftemperaturen gelegt worden. Für andere Optimierungen, wie hier beispielsweise zur Reduzierung der Wärmeverluste, müsste die Optimierung wieder um zusätzliche Restriktionen wie die Begrenzung der Leistungsgradienten ergänzt werden.

7.4. Optimierung von Fernwärmenetz 4

Für die Optimierung von Fernwärmenetz 4 wird das gleiche Vorgehen wie bisher verwendet. Nicht alle Eingangsgrößen der Netzmodelle sind dabei in die Optimierung integriert. Eine grundlegende Optimierung der Leistungen und Vorlauftemperaturen ist dennoch möglich und wird im Anhang A.2 beschrieben.

7.5. Diskussion der Optimierungsergebnisse

Die Ergebnisse der Optimierungen und einige Limitierungen werden in diesem Kapitel diskutiert. Die Optimierung mit dem Ziel der Einhaltung wirtschaftlich optimaler Leistungs- und Vorlauftemperatur-sollwerte ist für vier Fernwärmenetze an den Beispieltagen prinzipiell lauffähig. Beispielnetz 1 wurde in Kapitel 7.1 als Grundlage verwendet, um die Optimierung unter Verwendung von neuronalen Netzen und Systemidentifikationsmodellen zu testen. Zusätzlich wurde in Kapitel 7.1.2 beschrieben, wie auch eine Wärmeerzeugungsanlage vereinfacht modelliert und optimiert werden kann und in Kapitel 7.1.3 eine instationäre Optimierung vorgestellt.

Anhand von Netz 2 konnte in Kapitel 7.2 ein einfaches Beispiel der verteilten Wärmeeinspeisung untersucht werden. Sollwerte für Leistungen und Vorlauftemperaturen der Wärmeerzeugungsanlagen können bei Verletzung von Grenzwerten möglichst nah am wirtschaftlichen Optimum umgeplant werden. Die Optimierung stellt so sicher, dass in einem 24-Stunden-Prognosezeitraum alle thermohydraulischen Grenzwerte eingehalten werden. Für das Netz werden mit dem gradientenbasierten Optimierer SLSQP gute Ergebnisse bei Rechenzeiten von etwa 1 s erreicht.

Zuletzt wurde die erarbeitete Methodik auf das komplexe Fernwärmenetz 3 mit mehreren verteilten Wärmeerzeugungsanlagen und zusätzlichen Nebenbedingungen durch eine Geothermieanlage und eine DEA angewandt. In diesem Fall sind aufgrund der höheren Anzahl an Optimierungsvariablen globale Optimierungsansätze wie der Algorithmus *differential evolution* notwendig. Dennoch können gute Ergebnisse bei geringen Rechenzeiten erreicht werden. An Sommertagen mit niedrigem Wärmeverbrauch treten in den Optimierungen noch vereinzelt Probleme auf, welche schon in der Modellbildung und -auswertung ab Kapitel 6.1 beschrieben wurden. Validierungsrechnungen in *SIR 3S*[®] bestätigen jedoch für alle Beispieltage plausible Optimierungsergebnisse. Für die Netze 2 und 3 wurden stationäre Optimierungen zudem durch instationäre Nachrechnungen in *SIR 3S*[®] validiert.

Insgesamt konnte das in dieser Arbeit erarbeitete Optimierungskonzept an mehreren Fernwärmenetze getestet werden. Eine stationäre Optimierung, welche neuronale Netze als Simulationsmodell einsetzt, ist so in der Lage, auch komplexe Netze mit verteilten Wärmeeinspeisungen und zusätzlichen Anforderungen durch Geothermie- und Druckerhöhungs- und Speicheranlagen schnell zu optimieren. Die Limitierungen der einzelnen Netzmodelle und Optimierungen wurden in den jeweiligen Kapiteln aufgeführt.

Hier soll noch zusammenfassend auf Abweichungen zwischen den Optimierungsergebnissen und der Realität eingegangen werden und wie diese im Realfall behandelt werden können. Zunächst kommen Abweichungen durch die bereits in Kapitel 6.5 aufgeführten Modellfehler zustande. So sind Berechnungsfehler zwischen *SIR 3S*[®] und Messwerten sowie zwischen neuronalen Netzen und *SIR 3S*[®] vorhanden. Die Ergebnisse der neuronalen Netze werden dabei durch die Vorgabe der Rücklauftemperaturen deutlich verbessert. Für Netz 2 ist eine Prognose der Rücklauftemperaturen bereits möglich, welche aber ebenfalls nicht fehlerfrei ist.

Zudem kommt es zu Fehlern zwischen den in dieser Arbeit entwickelten stationären Simulationen und realen instationären Vorgängen. Die Optimierungsergebnisse in den vorherigen Kapiteln wurden in stationären und instationären *SIR 3S*[®] Simulationen verglichen. Zwischen den Simulationen ergeben sich nur geringe Abweichungen, welche dennoch beachtet werden müssen. Eine Temperaturspeicherung in den Vorlauftemperaturleitungen, um Lastspitzen abzufangen, kann mit den stationären Optimierungen nicht berücksichtigt werden. Durch das Betriebspersonal kann, wie in Kapitel 7.2.3 beschrieben, eine Erhöhung oder Absenkungen der Vorlauftemperaturen und Leistungen aber wie bisher durchgeführt werden. Folglich entstehen leichte Veränderungen der Temperaturverteilungen zwischen dem realen Fernwärmenetz und der Netzsimulation.

Eine weitere Herausforderung ist die Vorhersage der gesamten Einspeiseleistung durch die wirtschaftliche Optimierung. In den vorherigen Abschnitten dieser Arbeit wurde gezeigt, dass die 24-Stunden-Prognosen vom Vortag nicht immer genaue Ergebnisse liefern. Eine Möglichkeit wäre in diesem Fall jeweils die zum Zeitpunkt $t = 0$ der Optimierung gemessene Abweichung zwischen tatsächlicher Einspeiseleistung und Prognose für die nächsten 24 Stunden zu übernehmen. Es konnte für die Fernwärmenetze gezeigt werden, dass die Optimierung auch mit daraus folgenden voneinander abweichenden Leistungsprognosen und Leistungssollwerten funktioniert. Mittlerweile besteht bei den in dieser Arbeit behandelten Fernwärmenetzen die Möglichkeit, auf eine 15-Minuten-Onlineprognose zuzugreifen. Diese bestimmt alle 15 Minuten neu optimierte Sollwerte und führt auch die Wärmeleistungsprognose häufiger durch. Bei Verwendung dieser Prognose sollten die Startwerte für die Optimierung bereits besser zur realen Netzsituation passen und eine thermo-hydraulische Optimierung genauer und schneller möglich sein.

Die aufgeführten Abweichungen können in dem hier vorgestellten Konzept teilweise durch das Betriebspersonal korrigiert werden. Auch die unterlagerten Differenzdruck-, Druck- und Temperaturregelungen sind in der Lage, kleinere Abweichungen abzufangen. Insgesamt ist, wie bereits in Kapitel 6.5 für die neuronalen Netze beschrieben, bei einer Inbetriebnahme der Optimierung eine genaue Überwachung durch das Personal und wenn nötig, eine Nachbesserung der Modelle und Optimierung notwendig. Ein Zeitraum von einer gesamten Heizperiode von einem Jahr ist dabei erforderlich, um Simulationen und Optimierung für alle Netzzustände zu validieren. Während dieses Zeitraums muss das Zusammenspiel zwischen der Optimierung, dem Personal und der Regelung auf der Feldebene erprobt werden und sichergestellt werden, dass die Simulationen und Optimierungen für jegliche Ausnahmesituation robuste Sollwerte liefern.

Ein weiterer wichtiger Punkt ist die Berücksichtigung maximaler Vorlauftemperaturgradienten. Die Optimierung wird wie hier für die Netze 2 bis 4 programmiert, fortlaufend und in jedem Zeitschritt einzeln durchgeführt. Gradientenbegrenzungen von den vorherigen Zeitschritten werden in der Optimierung berücksichtigt. Diese Nebenbedingung ist also auch zeitabhängig. Theoretisch ist es dabei möglich, dass die Optimierung über den 24-Stunden-Horizont in Zustände läuft, in welchen die Temperaturgradienten nicht eingehalten werden können, da zurückliegende Zeitschritte ohne Eingriff nicht erneut optimiert werden können. In den durchgeführten Analysen ist eine solche Situation nicht aufgetreten. Dennoch muss dieser Punkt praktisch getestet werden. Alternativ bleibt die Möglichkeit, alle Zeitschritte gleichzeitig zu optimieren. In diesem Fall wäre nur eine einzige Optimierung mit einer deutlich höheren Anzahl an Variablen und Restriktionen notwendig, welche länger dauern würde, allerdings sicher zeitabhängige Restriktionen einhält.

Weiterhin soll auf die Optimierung der Wärmeverluste eingegangen werden. Es wurde dargestellt, dass eine Minimierung der Wärmeverluste zu einer dynamischen Anpassung der Vorlauftemperatur führt, welche im Vergleich von stationären und instationären *SIR 3S*[®] Simulationen deutliche Abweichungen verursacht. Folglich wurde eine instationäre Optimierung mit Systemidentifikationsmodellen grundlegend für Netz 1 getestet. Dabei konnte prinzipiell gezeigt werden, dass eine solche Optimierung möglich ist. Weitere Forschungsarbeiten können dieses Thema im Detail untersuchen. Auch für die Fernwärmenetze 2 bis 4 wurde eine Wärmeverlustminimierung untersucht, ist jedoch im erreichten Zustand nicht einsatzfähig.

8. Fazit und Ausblick

Der Betrieb von Fernwärmenetzen wird in Zukunft durch den Ausbau regenerativer Energien mit zunehmend kleineren, dezentral verteilten Wärmeerzeugungsanlagen komplexer. In der vorliegenden Arbeit wurde ein Automatisierungskonzept vorgestellt und theoretisch untersucht, welches die Möglichkeit bietet, Fernwärmenetze wirtschaftlich optimal unter Einhaltung aller thermo-hydraulischen Restriktionen zu betreiben.

Zunächst wurde der Stand der Technik der Optimierung und Regelung von Fernwärmenetzen dargelegt. Auf dieser Basis wurde vorgeschlagen, den Betrieb bestehender Fernwärmenetze mit wirtschaftlicher Optimierung um eine Ebene mit Netzsimulation und modellprädiktiver Regelung (MPC) zu erweitern. Die lineare wirtschaftliche Optimierung liefert Sollwerte für die Leistungen und Vorlauftemperaturen der im Netz verteilten Wärmeerzeugungsanlagen sowie eine Prognose der notwendigen Wärmeleistung. Die MPC verwendet diese Sollwerte als Zielgrößen und berechnet mittels nicht-linearer Netzsimulationen thermo-hydraulische Restriktionen wie minimale und maximale Drücke oder Temperaturen. Können Grenzwerte nicht eingehalten werden, plant die MPC die Zielgrößen um und ermittelt neue umsetzbare Leistungen und Vorlauftemperaturen möglichst nah am wirtschaftlichen Optimum. Eine vollständige Umsetzung einer MPC für komplexe Fernwärmenetze erfordert deutliche Änderungen an deren Betriebsweise. Daher wurde zunächst eine thermo-hydraulische Optimierung entwickelt, welche die Sollwerte an das Betriebspersonal und die Regelungen auf der Feldebene weitergeben soll und so die aktuelle Betriebsweise möglichst wenig verändert.

Mehrere Ansätze der Netzmodellierung wurden erprobt und schließlich vier Fernwärmenetze mit zunehmender Komplexität von der Firma 3S Consult GmbH in der Software *SIR 3S*[®] modelliert sowie mit Messdaten von Beispieltagen zu unterschiedlichen Jahreszeiten validiert. Die Ergebnisse von quasi-stationären Tagessimulationen zeigen eine gute Übereinstimmung mit den Messdaten, dauern allerdings je nach Komplexität der Netze zwischen einigen Sekunden und mehreren Minuten. Somit sind die physikalischen Netzmodelle für eine Anwendung in einer Optimierung oder MPC nicht geeignet. Anschließend wurden Ansätze des maschinellen Lernens angewandt, um die detaillierten Modelle zu vereinfachen und Rechenzeiten von stationären Simulationen zu reduzieren. Für das Training wurden Datensätze aus Ein- und Ausgangsdaten der *SIR 3S*[®] Simulationen anstelle von Messdaten verwendet. Dies hat den Vorteil, dass zukünftige Netzausbauzustände durch eine Anpassung der *SIR 3S*[®] Modelle abgebildet und untersucht werden können. Je nach Netzkomplexität sind einige Hunderttausend bis zu mehrere Millionen stationäre Simulationen notwendig, um gute Trainingsergebnisse zu erzielen. Von verschiedenen Ansätzen zeigen neuronale Netze die besten Ergebnisse und wurden für die vier Fernwärmenetze trainiert. Die Berechnungsdauer von Tagessimulationen mit 15-Minuten-Zeitschritten kann so auf weniger als eine Millisekunde bei guter Übereinstimmung mit *SIR 3S*[®] reduziert werden. Limitierungen wurden in den Ergebnisdiskussionen hervorgehoben. Spezifische Untersuchungen wurden bezüglich der Rücklauftemperaturen vorgenommen. Diese sind als Eingangsgröße der *SIR 3S*[®] Simulationen notwendig und führen zu verbesserten Berechnungsergebnissen bei Verwendung als Eingang der neuronalen Netze. Für Prognoserechnungen im Rahmen einer Optimierung sind die Temperaturen allerdings nicht bekannt. Daher wurde für Netz 2 zusätzlich erfolgreich getestet, die Rücklauftemperaturen mit neuronalen Netzen basierend auf Messdaten vorherzusagen.

Mit den vereinfachten stationären Modellen wurden Optimierungen für komplexe Fernwärmenetze inklusive erweiterter Anforderungen durch Geothermie-, Druckerhöhungs- und Speicheranlagen entwi-

ckelt. Zudem konnte gezeigt werden, dass die direkte Vorgabe wirtschaftlich optimaler Sollwerte zur Verletzung thermo-hydraulischer Grenzwerte führen kann. Die entwickelte Optimierung ist anschließend in der Lage, unter Beachtung der Leistungsprognose und der Grenzwerte eine Lösung möglichst nah am wirtschaftlichen Optimum zu finden. Einfache Netze werden in Sekunden, komplexere Netze in etwa einer Minute optimiert. Dabei sind für Optimierungsprobleme mit wenigen Variablen lokale gradientenbasierte Verfahren geeignet und finden schnell Lösungen. Für Probleme mit einer höheren Anzahl an Variablen sind hingegen globale stochastische Optimierungsansätze notwendig, welche längere Rechenzeiten benötigen. Des Weiteren wurde das Verhalten der Optimierung bei abweichenden Leistungsprognosen untersucht. Dies erhöht teils die Rechenzeiten, eine Lösung wird dennoch gefunden. Zudem wurden die stationären Optimierungsergebnisse mit thermisch instationären Simulationen in *SIR 3S*[®] validiert. Effekte wie die Nutzung der Vorlaufleitungen als Wärmespeicher werden mit stationären Optimierungen bisher nicht berücksichtigt. Ein Be- oder Entladen der Leitungen kann allerdings wie bisher vom Betriebspersonal durchgeführt werden.

Auch eine Minimierung der Wärmeverluste durch Absenkung der Vorlauftemperaturen wurde untersucht. Bei dynamisch angepassten Vorlauftemperaturen ergeben sich größere Abweichungen zwischen thermisch stationären und instationären Nachrechnungen in *SIR 3S*[®]. Daher wurde die Systemidentifikation für die Erstellung von vereinfachten, thermisch instationären Modellen verwendet. Durch nicht-lineare Modelle konnte der Zusammenhang zwischen Ein- und Ausgangsgrößen für Netz 1 erlernt werden. Die Berechnungsdauer für eine Tagessimulation mit 1-Minute-Zeitschritten liegt bei unter einer Sekunde. Anschließend wurde grundlegend eine instationäre Optimierung mit Systemidentifikationsmodellen getestet. Die Ergebnisse können durch weitere Forschungsarbeiten verbessert werden. Dabei kommen auch neuronale Netze für die Nachbildung instationärer Simulationen infrage.

Bisher wurde die Optimierungskomponente einer MPC basierend auf Simulationen entwickelt und erfolgreich getestet. Diese Arbeit weiterführende Untersuchungen sind die Anwendung der Optimierungsergebnisse im praktischen Einsatz, der Umgang mit entstehenden Abweichungen sowie Tests der notwendigen Rückkopplung zur wirtschaftlichen Optimierung. Zudem muss sichergestellt werden, dass neuronale Netze und Optimierungen unabhängig vom Betriebsfall stets eine Lösung finden. Dazu gehört der Umgang mit Ausnahmesituationen wie der Absperrung einzelner Versorgungsgebiete und baulichen Änderungen an den Netzen. In dieser Arbeit wurden die Netze vollständig modelliert. Bei Änderungen können diese durch eine Anpassung der Modelle in *SIR 3S*[®] sowie dem erneuten Erzeugen von Trainingsdaten und Training neuronaler Netze berücksichtigt werden. Weitere Forschungsarbeiten können untersuchen, dieses Vorgehen zu vereinfachen. Zudem sollte eine Validierung der erstellten Modelle und Optimierungen während einer Heizperiode durchgeführt werden. In dieser Zeit können neuronale Netze und Optimierungsalgorithmen kontinuierlich verbessert werden. Auch der Einsatz in einer MPC ist anschließend denkbar. Dafür müssen eine direkte Aufschaltung der optimierten Leistungs- und Vorlauftemperaturwerte auf die Wärmeerzeugungsanlagen erfolgen und die Rechenzeiten gering sein. Für die Fernwärmenetze 1 und 2 sind Optimierungen bereits in unter einer Sekunde möglich. Für Netz 3 und 4 sind die Optimierungszeiten von unter einer Minute bezogen auf die trägen thermo-hydraulischen Systeme voraussichtlich ebenfalls ausreichend schnell.

Zusammenfassend wurde in der vorliegenden Arbeit ein Automatisierungskonzept für Fernwärmenetze basierend auf neuronalen Netzen und Systemidentifikationsmodellen entwickelt und mit detaillierten physikalischen Netzsimulationen getestet. Das Konzept erweitert eine wirtschaftliche Optimierung der Leistungs- und Vorlauftemperatursollwerte um eine hydraulische und thermische Optimierung, welche die Einhaltung thermo-hydraulischer Restriktionen sicherstellt. Das Vorgehen wurde für zunehmend komplexe Fernwärmenetze mit verteilter Wärmeerzeugung und zusätzlichen Anforderungen durch Geothermie-, Druckerhöhungs- und Speicheranlagen untersucht und zeigt sowohl in stationären als auch instationären Nachrechnungen mit einer physikalischen Netzsimulation gute Ergebnisse. Tests in der praktischen Anwendung können die Modelle und Optimierungen weiter verbessern. Zudem wurden Limitierungen und Ansätze für weiterführende wissenschaftliche Untersuchungen erläutert.

A. Anhang

A.1. Weitere Informationen zur Modellbildung und -auswertung

In diesem Anhang wird der Programmcode der neuronalen Netze beschrieben. Anschließend wird die Auswahl der Netzstruktur pro Fernwärmenetz beschrieben und die Modelle detailliert verglichen zu *SIR 3S*[®] ausgewertet.

A.1.1. Programmcode: Neuronale Netze

PyTorch Neuronales Netz Der folgende Programmcode beschreibt ein neuronales Netz in der *Python* Bibliothek *PyTorch*. In den Variablen *n_inputs* und *n_outputs* werden die Anzahl der Eingänge und Ausgänge des Netzes übergeben. *layerNodes* enthält einen Vektor mit der Länge der Anzahl versteckter Schichten und den Anzahlen der Neuronen pro Schicht. Die Schichten werden als *Linear-Layer* in den Zeilen 14, 17 und 20 miteinander verbunden. Für die Aktivierung der Neuronen wird die Funktion *Swish* (hier *SiLU*) verwendet. Alle Schichten und Aktivierungsfunktionen werden in einer Liste aneinandergereiht und letztendlich durch einen *Sequential-Container* in Zeile 23 miteinander verbunden.

```
1 import torch
2 from torch import nn
3
4 # class for PyTorch neural network
5 class PyTorch_MN(nn.Module):
6     # n_inputs = number of input nodes
7     # n_outputs = number of output nodes
8     # layerNodes = list with number of nodes per hidden layer,
9     # e.g. [110, 88, 44]
10    def __init__(self, n_inputs, n_outputs, layerNodes):
11        super().__init__()
12        # connect input layer with first hidden layer. Swish as
13        # activation function:
14        modules = [nn.Linear(n_inputs, layerNodes[0]), nn.SiLU()]
15        # connect each additional layer:
16        for i in range(1, len(layerNodes)):
17            modules.append(nn.Linear(layerNodes[i-1], layerNodes[i]))
18            modules.append(nn.SiLU())
19        # connect last hidden layer with output layer:
20        modules.append(nn.Linear(layerNodes[-1], n_outputs))
21        modules.append(nn.Identity(n_outputs))
22        # add all modules to a Sequential container:
23        self.layers = nn.Sequential(*modules)
24
25    # forward function to calculate outputs of neural network:
26    def forward(self, x):
27        return self.layers(x)
```

Datensatz für das Training in *PyTorch* Hier wird ein Datensatz der Trainingsdaten beschrieben, welcher später während des Trainings in *PyTorch* von einem *DataLoader* verwendet wird, um Daten in *Batches* zu laden. Während der Initialisierung ab Zeile 5 werden die Länge eines Datensatzes bestimmt und die Ein- und Ausgangsdaten (*features* und *targets*) gespeichert. Durch die Funktion *getitem* in Zeile 11 wird jeweils eine Kombination aus Eingangs- und Ausgangsdaten als Tupel ausgegeben.

```

1 from torch.utils.data import Dataset
2
3 # Class for dataset. Can be used by PyTorch DataLoader during training:
4 class CustomDataset(Dataset):
5     def __init__(self, features, targets):
6         # sets number of samples, features and targets tensor
7         self.n_samples = targets.shape[0]
8         self.features = torch.from_numpy(features).float()
9         self.targets = torch.from_numpy(targets).float()
10
11     def __getitem__(self, index):
12         # returns all features and targets for given index:
13         return self.features[index, :], self.targets[index, :]
14
15     def __len__(self):
16         # get length of dataset:
17         return self.n_samples

```

Training in *PyTorch* Der folgende Programmcode beschreibt das Training eines neuronalen Netzes in *PyTorch*. Die Bibliotheken *numpy*, *PyTorch* und *scikit-learn* müssen dafür importiert werden. Zusätzlich sind die vorab erläuterten Klassen für das neuronale Netz *PyTorch_NN* und für den Datensatz *CustomDataset* notwendig. Als Eingangsdaten werden der Funktion die Ein- und Ausgangsdaten sowie ein *Dictionary* mit Parametern übergeben. Dieses enthält die Anzahl der Schichten, die Neuronen pro Schicht und die Anzahl der Epochen für das Training. Anschließend wird in Zeile 18 festgelegt, ob eine *GPU* oder *CPU* für das Training verwendet wird. Danach werden die Daten in Zeile 24 in Trainings- und Validierungsdaten unterteilt. 20% der Daten werden bereits vor dem Training als Testdaten entnommen. Die zufällige Aufteilung der Daten erfolgt durch die Funktion *train_test_split* der Bibliothek *scikit-learn*. Die Trainings- und Validierungsdaten werden in Zeile 27 und 28 in den im vorherigen Absatz erläuterten Datensätzen gespeichert. *DataLoaders* für beide Datensätze werden ab Zeile 31 erstellt. Diese laden die Daten während des Trainings in *Batches* von jeweils 256 Ein- und Ausgangsdaten. Anschließend wird in Zeile 40 ein Objekt des in diesem Kapitel beschriebenen neuronalen Netzes erstellt. Die Gewichte des Netzes können in Zeile 43 durch die Funktion *initialize_weights* initialisiert werden. Die Zeile kann auch auskommentiert werden, um die Standardinitialisierung von *PyTorch* zu verwenden. Die Funktion *initialize_weights* wird ab Zeile 121 beschrieben. *Loss*-Funktion und der Optimierer werden in Zeile 46 und 47 gewählt. Anschließend kann eine abnehmende Lernrate durch einen *learning rate scheduler* eingestellt werden.

Das Training selbst beginnt in Zeile 61 mit einer Schleife pro Lernepoche. Das Training wird *Batch*-weise ab Zeile 67 durchgeführt. Für jedes *Batch* werden zunächst die Gradienten auf 0 gesetzt, der *Loss* für ein *Batch* berechnet und in jedem Schritt aufaddiert sowie letztendlich die Optimierung der *Loss*-Funktion durch einen Schritt per *Backpropagation* durchgeführt. In dieser Optimierung werden die Gradienten der zuvor berechneten *Loss*-Funktion bestimmt und Gewichte entsprechend der steilsten Abstiegsrichtung verändert. Die Gewichte werden so *Batch* für *Batch* aktualisiert und die *Loss*-Funktion minimiert. Ab Zeile 87 wird nach dem Training jeder Epoche die *Loss*-Funktion der Validierungsdaten berechnet. Hier wird ebenfalls *Batch*-weise vorgegangen, um Arbeitsspeicher zu sparen. Ab Zeile 103 wird noch die grafische Darstellung einer Lernkurve programmiert. Das Training wird so für jede Epoche wiederholt. Durch die Lernkurve kann überprüft werden, ob es zu einer Abweichung zwischen Trainings- und Validierungsloss kommt. Dies würde auf eine Überanpassung an die

Trainingsdaten hindeuten. Theoretisch könnte hier auch das sogenannte *Early-Stopping* implementiert werden. So kann das Training abgebrochen werden, falls sich der Validierungsloss im Vergleich zum Trainingsloss verschlechtert. Da dies für die hier verwendeten Daten der Fernwärmenetze bisher kein Problem darstellt, wird das Training immer für die volle Anzahl an Epochen durchgeführt.

```
1 import numpy as np
2 import torch
3 from torch import nn
4 from torch.autograd import Variable
5
6 from sklearn.model_selection import train_test_split
7
8 import PyTorch_NN
9 import CustomDataset
10
11 # x_train = features
12 # y_train = targets
13 # params = Dictionary with layers per node and number of epochs for training,
14 # e.g. params = {'LayersNodes': [110, 88, 44], 'Epochs': 1000}
15 def train_PyTorch(x_train, y_train, params):
16     # use GPU if available:
17     device = "cuda" if torch.cuda.is_available() else "cpu"
18     print(f"Using {device} device")
19
20     # x_train, y_train = 80 % of data / 20 % already split as test data
21     # split x_train, y_train into 75 % training, 25 % validation for
22     # 60 / 20 / 20 split
23     x_train, x_val, y_train, y_val = \
24     train_test_split(x_train, y_train, test_size = 0.25)
25
26     # Create training and test dataset:
27     train_dataset = CustomDataset(x_train, y_train)
28     test_dataset = CustomDataset(x_val, y_val)
29
30     # Create data loader for batch training:
31     train_loader = torch.utils.data.DataLoader(dataset = train_dataset, \
32         batch_size=256, num_workers=0, shuffle=True, pin_memory=True)
33
34     # create data loader for testdata evaluation:
35     test_loader = torch.utils.data.DataLoader(dataset = test_dataset, \
36         batch_size=256, num_workers=0, shuffle=True, pin_memory=True)
37
38     # Create PyTorch Neural Network object:
39     n_inputs, n_outputs = x_train.shape[1], y_train.shape[1]
40     model = PyTorch_NN(n_inputs, n_outputs, params['LayersNodes']).to(device)
41
42     # initialize weights of neural network with custom function (optional):
43     model.apply(initialize_weights)
44
45     # Define the loss function, optimizer and learning rate (lr, 1e-3 = standard)
46     loss_function = nn.L1Loss() # L1Loss = MAE / MSELoss = MSE
47     optimizer = torch.optim.Adam(model.parameters(), lr=1e-3)
48
49     # learning rate scheduler, define learning rate decay by "gamma" every
50     # "step_size" epochs (optional):
51     # scheduler = torch.optim.lr_scheduler.StepLR(optimizer, step_size=200, gamma=0.5)
52
53     # load number of epochs from "params" input
54     epochs = params['Epochs']
55
56     # save training and validation losses in numpy lists:
57     trainlosses = np.zeros(epochs)
```

```

58 validationlosses = np.zeros(epochs)
59
60 # Run the training loop
61 for epoch in range(0, epochs):
62     # Print epoch
63     print(f'Starting epoch {epoch+1}')
64     model.train() # put model in training mode
65     train_loss_running = 0
66     # batch learning:
67     for batch, (features, targets) in enumerate(train_loader):
68         features = features.to(device) # send data to GPU / CPU
69         targets = targets.to(device) # send data to GPU / CPU
70
71         # Zero the gradients:
72         for param in model.parameters():
73             param.grad = None
74
75         prediction = model(features)
76
77         # Compute loss of batch
78         loss = loss_function(prediction, targets)
79         train_loss_running += float(loss)
80
81         loss.backward() # Back propagation
82         optimizer.step() # Perform optimization
83
84     # Compute loss over complete training data for epoch:
85     trainlosses[epoch] = train_loss_running / len(train_loader)
86
87     model.eval() # model in evaluation mode
88     # Compute loss over complete validation data for epoch:
89     # Don't track gradient for validation loss
90     with torch.no_grad():
91         test_loss_running = 0
92         # Iterate over the test data and generate predictions
93         for batch, (features, targets) in enumerate(test_loader):
94             features = features.to(device) # send data to GPU / CPU
95             targets = targets.to(device) # send data to GPU / CPU
96             prediction = model(features)
97             # Compute loss of batch
98             loss = loss_function(prediction, targets)
99             test_loss_running += float(loss)
100         validationlosses[epoch] = test_loss_running / len(test_loader)
101
102     # Plot learning curve:
103     if (epoch+1) % 100 == 0:
104         # plot and show learning process
105         plt.cla()
106         plt.plot(range(0, epoch+1), trainlosses[0:epoch+1])
107         plt.plot(range(0, epoch+1), validationlosses[0:epoch+1])
108         plt.legend(['Train Loss', 'Validation Loss'])
109         plt.xlim([0, epochs])
110         plt.grid()
111
112     # learning rate decay, optional:
113     # scheduler.step()
114
115 # Process is complete.
116 # transfer model to cpu
117 model.to("cpu")
118 return model
119

```

```

120 # function to initialize weights of neural network:
121 def initialize_weights(m):
122     if isinstance(m, nn.Linear):
123         nn.init.kaiming_uniform_(m.weight.data, nn.init.calculate_gain('relu'))
124         nn.init.constant_(m.bias.data, 0)

```

A.1.2. Modellvalidierung Netz 1

Hier werden weitere Informationen zur Auswahl des neuronalen Netzes für Fernwärmenetz 1 gegeben. Die Modelle haben drei Eingänge und elf Ausgänge. Die Eingänge sind die Wärmeabnahme, die Vorlauftemperatur und der Vorlaufdruck. Die Ausgänge sind der minimale Differenzdruck, die Pumpendrehzahl und -leistung, der Rücklauf-, maximale Vorlauf- und minimale Rücklaufdruck, die Umwälzmenge, die minimale und maximale Strömungsgeschwindigkeit, der Wärmeverlust und die minimale Vorlauftemperatur. Die Anzahl der Neuronen wird jeweils als Vielfaches der Anzahl der Ausgänge gewählt, die Anzahl der Schichten zwischen 1 und 4 variiert. Das Training wird mit *PyTorch* und den folgenden Parametern durchgeführt:

- 60%/20%/20% Unterteilung zwischen Trainings-, Validierungs- und Testdaten und Normierung der Ein- und Ausgangsdaten zwischen 0 und 1
- Optimierer: *Adam*
- Loss-Funktion: MAE
- Aktivierungsfunktion: *Swish*
- Epochen: 100

In der folgenden Tabelle A.1 ist die Auswertung der trainierten Modelle dargestellt. Für die Ausgangsgröße minimaler Differenzdruck Δp ist der MAPE und MAE angegeben. Zusätzlich sind die skalierten Werte über alle Ausgangsgrößen MAE, MSE, RMSE und R^2 aufgelistet. Des Weiteren ist ein Bewertungskriterium *Score* in der Tabelle angegeben. Hier wurden für jedes Modell 1 bis 5 Punkte entsprechend Tabelle A.2 vergeben. Die nach dieser Bewertung ausgewählten Modelle sind in Tabelle A.1 grün markiert. Es wird jeweils das beste neuronale Netz mit einer, zwei und drei Schichten gewählt. Bei mehreren Netzen mit gleicher Bewertung wird das Modell mit der geringeren Anzahl an Neuronen selektiert.

Tabelle A.1.: Bewertung neuronaler Netze mit Testdaten für Fernwärmenetz 1 nach 100 Epochen Training.

Neuronen pro Schicht	MAPE Δp (%)	MAE Δp (bar)	MAE	MSE	RMSE	R^2	Score
44	166,455	0,067	0,00676	0,00031	0,01212	0,99506	6
88	41,753	0,061	0,00616	0,00030	0,01173	0,99531	6
132	77,150	0,054	0,00565	0,00029	0,01125	0,99534	6
176	48,762	0,051	0,00535	0,00024	0,01044	0,99606	6
44/11	53,047	0,040	0,00441	0,00024	0,01000	0,99608	6
44/44	10,450	0,027	0,00359	0,00020	0,00873	0,99664	15
44/88	5,988	0,019	0,00322	0,00020	0,00840	0,99673	20
88/11	39,321	0,030	0,00359	0,00024	0,00901	0,99639	10
88/44	10,494	0,020	0,00334	0,00020	0,00849	0,99681	18
88/88	5,467	0,017	0,00284	0,00019	0,00790	0,99705	25
132/11	14,160	0,025	0,00351	0,00019	0,00842	0,99691	18

Tabelle A.1 fortgesetzt.

Neuronen pro Schicht	<i>MAPE</i> Δp (%)	<i>MAE</i> Δp (bar)	<i>MAE</i>	<i>MSE</i>	<i>RMSE</i>	R^2	Score
132/44	9,670	0,015	0,00266	0,00016	0,00741	0,99725	28
132/88	7,760	0,018	0,00297	0,00021	0,00847	0,99657	18
176/11	54,830	0,042	0,00452	0,00026	0,01027	0,99588	6
176/44	9,478	0,018	0,00279	0,00019	0,00778	0,99702	23
176/88	8,265	0,017	0,00286	0,00018	0,00791	0,99709	24
44/44/11	5,290	0,016	0,00292	0,00021	0,00825	0,99669	21
44/44/44	5,598	0,014	0,00270	0,00019	0,00772	0,99699	25
44/44/88	11,032	0,015	0,00299	0,00019	0,00810	0,99697	22
44/88/11	9,958	0,022	0,00354	0,00020	0,00858	0,99680	19
44/88/44	7,802	0,014	0,00283	0,00019	0,00805	0,99692	24
44/88/88	38,845	0,024	0,00346	0,00022	0,00863	0,99656	12
88/44/11	5,314	0,016	0,00293	0,00019	0,00779	0,99702	24
88/44/44	6,150	0,016	0,00269	0,00018	0,00774	0,99703	25
88/44/88	8,354	0,014	0,00266	0,00018	0,00781	0,99714	25
88/88/11	33,484	0,021	0,00319	0,00018	0,00785	0,99722	21
88/88/44	6,135	0,017	0,00347	0,00017	0,00828	0,99734	24
88/88/88	7,589	0,015	0,00330	0,00021	0,00859	0,99671	19
132/44/11	13,838	0,015	0,00304	0,00018	0,00799	0,99717	24
132/44/44	13,377	0,017	0,00291	0,00022	0,00841	0,99668	18
132/44/88	10,655	0,018	0,00280	0,00021	0,00791	0,99698	21
132/88/11	10,484	0,019	0,00342	0,00019	0,00841	0,99701	19
132/88/44	10,196	0,015	0,00302	0,00019	0,00809	0,99732	23
132/88/88	7,983	0,013	0,00296	0,00016	0,00772	0,99748	27
176/44/11	5,803	0,015	0,00292	0,00021	0,00835	0,99679	21
176/44/44	8,959	0,013	0,00280	0,00019	0,00772	0,99727	25
176/44/88	5,344	0,012	0,00255	0,00015	0,00721	0,99767	30
176/88/11	7,785	0,015	0,00314	0,00017	0,00789	0,99731	25
176/88/44	20,731	0,017	0,00292	0,00019	0,00767	0,99738	22
176/88/88	8,717	0,013	0,00257	0,00017	0,00739	0,99747	28
44/44/44/11	7,555	0,016	0,00259	0,00022	0,00808	0,99661	20
44/44/88/11	10,439	0,016	0,00280	0,00020	0,00794	0,99694	22
44/88/44/11	8,891	0,016	0,00301	0,00018	0,00799	0,99704	23
44/88/88/11	7,255	0,014	0,00284	0,00019	0,00789	0,99711	24
88/44/44/11	12,953	0,017	0,00308	0,00018	0,00797	0,99721	24
88/44/88/11	6,744	0,018	0,00341	0,00020	0,00855	0,99689	19
88/88/44/11	7,598	0,015	0,00306	0,00018	0,00802	0,99722	25
88/88/88/11	9,961	0,016	0,00283	0,00019	0,00784	0,99713	24
132/44/44/11	5,659	0,014	0,00273	0,00018	0,00752	0,99739	28
132/44/88/11	7,387	0,015	0,00321	0,00021	0,00834	0,99690	21
132/88/44/11	9,165	0,014	0,00265	0,00019	0,00765	0,99716	26
132/88/88/11	10,525	0,013	0,00261	0,00019	0,00782	0,99700	23
176/44/44/11	14,334	0,015	0,00282	0,00020	0,00798	0,99699	22
176/44/88/11	9,147	0,016	0,00287	0,00020	0,00787	0,99718	24
176/88/44/11	11,167	0,014	0,00244	0,00017	0,00723	0,99748	27

Tabelle A.2.: Bewertungsscore neuronaler Netze für Fernwärmenetz 1 in Tabelle A.1.

	<i>MAPE</i> Δp (%)	<i>MAE</i> Δp (bar)	<i>MAE</i>	<i>MSE</i>	<i>RMSE</i>	R^2
Minimum	5,226	0,012	0,00244	0,00015	0,00721	0,99506
Maximum	166,455	0,067	0,00676	0,00031	0,01212	0,99767
Punkte						
5	< 6,00	< 0,018	< 0,00287	< 0,00017	< 0,00770	> 0,99741
4	< 10,00	< 0,023	< 0,00330	< 0,00018	< 0,00819	> 0,99715
3	< 15,00	< 0,028	< 0,00374	< 0,00020	< 0,00868	< 0,99689
2	< 20,00	< 0,034	< 0,00417	< 0,00022	< 0,00917	> 0,99662
1	Rest	Rest	Rest	Rest	Rest	Rest

Anschließend wird das Training einiger neuronaler Netze mit 1000 Epochen durchgeführt. Die Ergebnisse dieser Berechnungen sind in Tabelle A.3 dargestellt. Die gleichen Bewertungskriterien wie zuvor werden verwendet. Die *Score* wird nach Tabelle A.4 vergeben und wieder die besten Netze mit zwei, drei und vier Schichten ausgewählt.

Tabelle A.3.: Bewertung neuronaler Netze mit Testdaten für Fernwärmenetz 1 nach 1000 Epochen Training.

Neuronen pro Schicht	<i>MAPE</i> Δp (%)	<i>MAE</i> Δp (bar)	<i>MAE</i>	<i>MSE</i>	<i>RMSE</i>	R^2	Score
22	56,573	0,030	0,00383	0,00021	0,00915	0,99644	6
66	22,641	0,027	0,00344	0,00024	0,00902	0,99609	6
110	11,099	0,026	0,00325	0,00022	0,00860	0,99649	6
22/22	10,694	0,015	0,00226	0,00020	0,00746	0,99716	10
22/44	14,739	0,013	0,00220	0,00016	0,00689	0,99774	14
22/88	6,914	0,016	0,00239	0,00018	0,00724	0,99754	11
66/22	4,816	0,013	0,00225	0,00018	0,00735	0,99739	15
66/44	8,679	0,013	0,00234	0,00017	0,00728	0,99773	13
66/88	6,506	0,012	0,00216	0,00016	0,00682	0,99790	18
110/22	12,334	0,013	0,00223	0,00015	0,00671	0,99803	17
110/44	4,519	0,012	0,00228	0,00016	0,00703	0,99779	18
110/88	10,873	0,013	0,00213	0,00018	0,00702	0,99765	14
22/44/22	4,829	0,013	0,00207	0,00018	0,00709	0,99763	18
22/44/44	10,034	0,012	0,00208	0,00015	0,00668	0,99797	18
22/44/88	6,542	0,013	0,00227	0,00016	0,00687	0,99790	17
22/88/22	4,516	0,012	0,00213	0,00016	0,00694	0,99787	21
22/88/44	5,256	0,012	0,00216	0,00011	0,00621	0,99838	26
22/88/88	13,001	0,012	0,00195	0,00017	0,00662	0,99782	17
66/44/22	5,382	0,011	0,00197	0,00015	0,00637	0,99807	25
66/44/44	11,882	0,013	0,00220	0,00014	0,00644	0,99832	20
66/44/88	5,295	0,012	0,00220	0,00016	0,00683	0,99801	18
66/88/22	7,337	0,011	0,00212	0,00016	0,00676	0,99792	19
66/88/44	6,788	0,011	0,00211	0,00016	0,00678	0,99792	19
66/88/88	6,016	0,012	0,00219	0,00017	0,00688	0,99784	16
110/44/22	4,260	0,012	0,00209	0,00018	0,00704	0,99762	18
110/44/44	5,221	0,012	0,00199	0,00017	0,00664	0,99783	19
110/44/88	4,837	0,012	0,00197	0,00017	0,00652	0,99798	21

Tabelle A.3 fortgesetzt.

Neuronen pro Schicht	<i>MAPE</i> Δp (%)	<i>MAE</i> Δp (bar)	<i>MAE</i>	<i>MSE</i>	<i>RMSE</i>	R^2	Score
110/88/22	9,000	0,012	0,00192	0,00018	0,00682	0,99772	16
110/88/44	4,824	0,011	0,00202	0,00014	0,00615	0,99832	27
110/88/88	7,935	0,014	0,00206	0,00018	0,00681	0,99774	16
22/44/44/22	5,910	0,010	0,00176	0,00012	0,00573	0,99849	29
22/44/44/44	5,361	0,013	0,00215	0,00018	0,00694	0,99787	18
22/44/88/22	4,363	0,012	0,00221	0,00016	0,00686	0,99792	19
22/44/88/44	5,760	0,012	0,00209	0,00017	0,00672	0,99787	19
22/88/44/22	8,274	0,013	0,00209	0,00016	0,00673	0,99787	18
22/88/44/44	7,214	0,012	0,00224	0,00016	0,00673	0,99807	19
22/88/88/22	5,673	0,012	0,00220	0,00015	0,00652	0,99815	21
22/88/88/44	7,373	0,011	0,00210	0,00015	0,00652	0,99816	22
66/44/44/22	5,517	0,012	0,00196	0,00017	0,00660	0,99790	20
66/44/44/44	10,406	0,012	0,00206	0,00016	0,00638	0,99811	19
66/44/88/22	4,436	0,015	0,00222	0,00018	0,00668	0,99784	18
66/44/88/44	7,971	0,011	0,00207	0,00015	0,00651	0,99816	22
66/88/44/22	6,662	0,013	0,00212	0,00018	0,00688	0,99783	17
66/88/44/44	5,445	0,010	0,00195	0,00012	0,00586	0,99854	29
66/88/88/22	7,018	0,015	0,00221	0,00018	0,00685	0,99784	15
66/88/88/44	5,670	0,011	0,00201	0,00013	0,00613	0,99834	26
110/44/44/22	7,113	0,011	0,00183	0,00014	0,00599	0,99832	27
110/44/44/44	5,008	0,012	0,00184	0,00016	0,00624	0,99805	23
110/44/88/22	10,756	0,014	0,00211	0,00016	0,00651	0,99809	17
110/44/88/44	6,829	0,012	0,00220	0,00017	0,00683	0,99789	16
110/88/44/22	5,065	0,012	0,00196	0,00014	0,00632	0,99822	25
110/88/44/44	4,313	0,010	0,00205	0,00014	0,00637	0,99831	27
110/88/88/22	7,947	0,012	0,00232	0,00015	0,00673	0,99818	21
110/88/88/44	5,417	0,011	0,00196	0,00016	0,00640	0,99805	24

Tabelle A.4.: Bewertungsscore neuronaler Netze für Fernwärmenetz 1 in Tabelle A.3.

	<i>MAPE</i> Δp (%)	<i>MAE</i> Δp (bar)	<i>MAE</i>	<i>MSE</i>	<i>RMSE</i>	R^2
Minimum	4,260	0,010	0,00176	0,00011	0,00573	0,99609
Maximum	56,573	0,030	0,00383	0,00024	0,00915	0,99854
Punkte						
5	< 5,00	< 0,012	< 0,00197	< 0,00013	< 0,00607	> 0,99829
4	< 6,00	< 0,014	< 0,00218	< 0,00014	< 0,00641	> 0,99805
3	< 8,00	< 0,016	< 0,00239	< 0,00015	< 0,00675	< 0,99780
2	< 10,00	< 0,018	< 0,00259	< 0,00016	< 0,00710	> 0,99756
1	Rest	Rest	Rest	Rest	Rest	Rest

In Kapitel 6.1.3 wird die Auswahl der letztendlich verwendeten neuronalen Netze beschrieben. Eine Architektur mit drei versteckten Schichten mit jeweils 110, 88 und 44 Neuronen wird gewählt und Modelle sowohl für das aggregierte als auch das vollständige Fernwärmenetz trainiert. Die folgenden Tabellen zeigen die maximalen relativen und maximalen absoluten Abweichungen für vier Beispieltage im Januar, April, Mai und Juli zunächst für ein neuronales Netz des aggregierten Fernwärmenetzes und anschließend für das vollständige Netz. Relative Abweichungen größer 3% sind rot hinterlegt.

Tabelle A.5.: Maximale absolute Abweichungen des neuronalen Netzes zu *SIR 3S*[®] für das aggregierte Netz 1.

Ausgangsgröße	Maximale absolute Abweichung zu <i>SIR 3S</i> [®]			
	Tag 1 (Januar)	Tag 2 (April)	Tag 3 (Juli)	Tag 4 (Mai)
Min. Δp (bar)	0,014	0,012	0,023	0,032
n_{Pumpe} (min ⁻¹)	5,412	3,192	5,786	0,757
P_{Pumpe} (kW)	12,729	5,091	5,218	1,044
p_{RL} (bar)	0,059	0,032	0,110	0,018
Max. p_{VL} (bar)	0,005	0,009	0,024	0,020
Min. p_{RL} (bar)	0,062	0,029	0,060	0,018
\dot{m} (t/h)	21,061	12,620	6,097	5,058
Max. v (m/s)	0,018	0,011	0,004	0,004
Min. v (m/s)	0,008	0,006	0,006	0,003
Wärmeverlust (MW)	0,005	0,005	0,019	0,014
Min. T_{VL} (°C)	0,069	0,853	5,987	4,171

Tabelle A.6.: Maximale relative Abweichungen des neuronalen Netzes zu *SIR 3S*[®] für das aggregierte Netz 1.

Ausgangsgröße	Maximale relative Abweichung zu <i>SIR 3S</i> [®] (%)			
	Tag 1 (Januar)	Tag 2 (April)	Tag 3 (Juli)	Tag 4 (Mai)
Min. Δp	1,101	0,784	0,698	1,030
n_{Pumpe}	0,457	0,360	0,820	0,100
P_{Pumpe}	1,392	1,301	3,784	0,645
p_{RL}	0,764	0,287	0,881	0,150
Max. p_{VL}	0,031	0,054	0,143	0,124
Min. p_{RL}	0,794	0,266	0,513	0,154
\dot{m}	0,583	0,532	0,897	0,678
Max. v	0,720	0,650	0,978	0,636
Min. v	1,014	1,187	5,875	1,494
Wärmeverlust	0,125	0,138	0,2267	0,415
Min. T_{VL}	0,061	0,975	9,206	5,183

Tabelle A.7.: Maximale absolute Abweichungen des neuronalen Netzes zu *SIR 3S*[®] für das vollständige Netz 1.

Ausgangsgröße	Maximale absolute Abweichung zu <i>SIR 3S</i> [®]			
	Tag 1 (Januar)	Tag 2 (April)	Tag 3 (Juli)	Tag 4 (Mai)
Min. Δp (bar)	0,016	2,568	399,228	120,044
n_{Pumpe} (min ⁻¹)	3,101	5,290	104,778	15,719
P_{Pumpe} (kW)	3,573	7,928	78,127	19,777
p_{RL} (bar)	0,028	0,024	0,884	0,065
Max. p_{VL} (bar)	0,007	0,013	0,069	0,018
Min. p_{RL} (bar)	0,030	0,057	0,916	0,041
\dot{m} (t/h)	9,520	101,188	324,796	218,076
Max. v (m/s)	0,010	0,108	0,202	0,173
Min. v (m/s)	0,006	0,005	0,155	0,040
Wärmeverlust (MW)	0,005	0,037	0,135	0,069
Min. T_{VL} (°C)	4,040	2,286	21,53	25,929

Tabelle A.8.: Maximale relative Abweichungen des neuronalen Netzes zu *SIR 3S*[®] für das vollständige Netz 1.

Ausgangsgröße	Maximale relative Abweichung zu <i>SIR 3S</i> [®] (%)			
	Tag 1 (Januar)	Tag 2 (April)	Tag 3 (Juli)	Tag 4 (Mai)
Min. Δp	1,483	804,693	427,785	109,808
n_{Pumpe}	0,317	0,709	12,763	2,125
P_{Pumpe}	0,827	4,048	34,568	10,614
p_{RL}	0,253	0,195	7,830	0,534
Max. p_{VL}	0,042	0,077	0,414	0,107
Min. p_{RL}	0,290	0,491	8,608	0,351
\dot{m}	0,311	8,624	32,933	20,218
Max. v	0,442	12,320	53,510	22,211
Min. v	0,906	2,219	56,416	18,298
Wärmeverlust	0,064	0,535	2,055	1,070
Min. T_{VL}	4,159	2,482	39,930	49,709

A.1.3. Modellvalidierung Netz 2

Hier werden Details der Auswahl des neuronalen Netzes für Fernwärmenetz 2 beschrieben. Die trainierten Modelle mit der Rücklaufftemperatur als Eingang haben sechs Eingänge und 26 Ausgänge. Die Eingänge sind die gesamte Wärmeeinspeisung, die Vorlauftemperatur des HKW, die Leistung und Vorlauftemperatur des HW sowie die Rücklaufftemperatur am HKW. Alle Ausgangsgrößen sind in Tabelle A.11 gegeben. Die Anzahl der Neuronen wird für das Training neuronaler Netze jeweils als Vielfaches der Anzahl der Ausgänge gewählt, die Anzahl der versteckten Schichten zwischen 1 und 4 variiert. Das Training wird mit *PyTorch* und den folgenden Parametern durchgeführt:

- 60 %/20 %/20 % Unterteilung zwischen Trainings-, Validierungs- und Testdaten und Normierung der Ein- und Ausgangsdaten zwischen 0 und 1
- Optimierer: *Adam*
- Loss-Funktion: MAE
- Aktivierungsfunktion: *Swish*
- Epochen: 100

In der folgenden Tabelle A.9 ist die Auswertung der trainierten Modelle dargestellt. Für die Ausgangsgrößen gesamter Massenstrom \dot{m} und minimaler Rücklaufdruck p_{RL} sind der MAE und / oder MAPE angegeben. Diese beiden Größen variieren zwischen den Modellen etwas stärker als die restlichen skalierten Werte über alle Ausgangsgrößen MAE, MSE, RMSE und R^2 . Zusätzlich ist ein Bewertungskriterium *Score* in der Tabelle angegeben. Hier werden für jedes Modell 1 bis 5 Punkte entsprechend Tabelle A.10 vergeben. Die nach dieser Bewertung ausgewählten Modelle sind in Tabelle A.9 grün markiert. Es wird jeweils das beste neuronale Netz mit einer, zwei, drei und vier Schichten gewählt. Bei mehreren Netzen mit gleicher Bewertung wird das Modell mit der geringeren Anzahl an Neuronen selektiert.

Tabelle A.9.: Bewertung neuronaler Netze mit Testdaten für Fernwärmenetz 2 nach 100 Epochen Training.

Neuronen pro Schicht	MAPE \dot{m} (%)	MAE \dot{m} (t/h)	MAPE p_{RL} (%)	MAE	MSE	RM-SE	R^2	Score
26	0,732	7,164	16,395	0,0120	0,00095	0,0221	0,94174	7
52	0,302	2,841	9,298	0,0098	0,00072	0,0189	0,95138	12
78	0,334	3,550	12,202	0,0099	0,00074	0,0190	0,95050	11
104	0,421	4,304	14,055	0,0092	0,00066	0,0182	0,95343	9
130	0,224	2,179	13,687	0,0087	0,00062	0,0174	0,95515	14
156	0,410	4,826	7,460	0,0090	0,00063	0,0176	0,95514	8
182	0,307	3,163	12,500	0,0085	0,00061	0,0172	0,95602	11
208	0,289	2,648	7,960	0,0089	0,00064	0,0176	0,95481	13
26/52	0,188	2,192	3,668	0,0048	0,00037	0,0125	0,96500	23
26/104	0,228	2,337	2,930	0,0043	0,00033	0,0114	0,96656	23
26/156	0,730	5,605	3,187	0,0043	0,00033	0,0113	0,96698	17
52/52	0,413	4,993	3,017	0,0043	0,00033	0,0116	0,96606	17
52/104	0,183	2,199	2,414	0,0038	0,00031	0,0108	0,96690	25
52/156	0,458	4,102	1,877	0,0041	0,00032	0,0112	0,96643	20
78/52	0,492	4,711	2,574	0,0043	0,00032	0,0114	0,96660	17
78/104	0,225	2,948	3,354	0,0039	0,00031	0,0109	0,96749	23
78/156	0,274	3,583	2,414	0,0041	0,00033	0,0113	0,96610	22
104/52	0,427	4,169	3,042	0,0039	0,00032	0,0110	0,96659	19

Tabelle A.9 fortgesetzt.

Neuronen pro Schicht	<i>MAPE</i> in (%)	<i>MAE</i> in (t/h)	<i>MAPE</i> PRL (%)	<i>MAE</i>	<i>MSE</i>	<i>RM- SE</i>	<i>R</i> ²	Score
104/104	0,216	2,657	2,200	0,0039	0,00031	0,0109	0,96683	24
104/156	0,409	4,615	3,470	0,0039	0,00030	0,0107	0,96807	18
130/52	0,181	2,495	3,871	0,0045	0,00032	0,0115	0,96651	23
130/104	0,243	3,841	2,611	0,0038	0,00031	0,0108	0,96747	21
130/156	0,574	5,473	3,221	0,0039	0,00029	0,0106	0,97722	18
156/52	0,379	4,211	2,015	0,0041	0,00032	0,0112	0,96657	20
156/104	0,675	5,321	2,738	0,0039	0,00029	0,0104	0,98878	20
156/156	0,347	3,040	5,632	0,0039	0,00031	0,0108	0,96758	19
182/52	0,426	4,336	2,257	0,0039	0,00031	0,0109	0,96701	20
182/104	0,415	5,742	1,995	0,0036	0,00027	0,0098	0,99198	26
182/156	0,270	2,885	3,686	0,0038	0,00030	0,0108	0,96731	23
208/52	0,212	2,540	2,967	0,0042	0,00033	0,0114	0,96633	23
208/104	0,254	2,744	2,248	0,0038	0,00031	0,0109	0,96684	24
208/156	0,257	3,095	2,442	0,0034	0,00028	0,0096	0,99200	28
26/52/52	0,212	3,021	2,228	0,0032	0,00030	0,0102	0,96678	24
26/52/104	0,357	4,472	3,303	0,0030	0,00028	0,0096	0,96706	22
26/104/52	0,244	4,197	2,097	0,0034	0,00030	0,0103	0,96669	23
26/104/104	0,197	2,259	1,767	0,0029	0,00027	0,0091	0,99248	32
26/156/52	0,245	4,440	3,272	0,0033	0,00030	0,0104	0,96674	22
26/156/104	0,167	2,635	1,947	0,0029	0,00027	0,0091	0,99257	32
52/52/52	0,218	2,721	1,866	0,0029	0,00029	0,0099	0,96703	26
52/52/104	0,168	2,596	1,721	0,0026	0,00027	0,0092	0,96843	29
52/104/52	0,134	2,229	1,728	0,0027	0,00029	0,0097	0,96648	28
52/104/104	0,244	3,425	1,999	0,0026	0,00026	0,0087	0,99281	31
52/156/52	0,139	1,740	1,626	0,0029	0,00028	0,0095	0,96717	28
52/156/104	0,168	2,153	1,863	0,0027	0,00028	0,0093	0,96801	29
78/52/52	0,132	2,003	1,762	0,0028	0,00029	0,0098	0,96688	28
78/52/104	0,177	2,262	1,275	0,0028	0,00029	0,0096	0,96703	27
78/104/52	0,185	2,611	1,737	0,0027	0,00029	0,0096	0,96743	28
78/104/104	0,195	2,748	2,591	0,0024	0,00025	0,0083	0,99294	33
78/156/52	0,159	2,403	1,695	0,0026	0,00027	0,0091	0,96725	29
78/156/104	0,192	3,042	1,483	0,0025	0,00025	0,0084	0,99300	33
104/52/52	0,291	5,372	3,405	0,0031	0,00029	0,0100	0,96701	21
104/52/104	0,309	3,743	3,228	0,0027	0,00027	0,0093	0,96789	25
104/104/26	0,199	2,757	2,046	0,0028	0,00028	0,0096	0,96712	27
104/104/52	0,122	1,692	1,861	0,0025	0,00025	0,0084	0,99292	35
104/104/104	0,136	2,315	1,338	0,0022	0,00024	0,0080	0,99308	34
104/156/26	0,224	3,742	2,482	0,0030	0,00029	0,0099	0,96659	24
104/156/52	0,216	2,404	2,930	0,0025	0,00026	0,0088	0,99250	31
104/156/104	0,181	2,376	1,557	0,0025	0,00026	0,0085	0,99254	33
130/52/52	0,152	2,214	2,093	0,0028	0,00029	0,0098	0,96673	28
130/52/104	0,397	4,420	1,461	0,0027	0,00026	0,0086	0,99272	28
130/104/52	0,314	4,044	2,400	0,0027	0,00025	0,0086	0,99192	29
130/104/104	0,220	2,814	1,457	0,0025	0,00025	0,0083	0,99275	33
130/156/52	0,160	1,996	1,471	0,0026	0,00027	0,0092	0,96741	30
130/156/104	0,222	4,400	1,314	0,0025	0,00026	0,0086	0,99237	30

Tabelle A.9 fortgesetzt.

Neuronen pro Schicht	<i>MAPE</i> in (%)	<i>MAE</i> in (t/h)	<i>MAPE</i> PRL (%)	<i>MAE</i>	<i>MSE</i>	<i>RM-SE</i>	<i>R</i> ²	Score
156/52/52	0,117	1,438	1,759	0,0026	0,00029	0,0095	0,96723	29
156/52/104	0,171	2,719	1,261	0,0025	0,00025	0,0084	0,99295	34
156/104/26	0,224	2,791	2,256	0,0031	0,00029	0,0100	0,96721	25
156/104/52	0,252	2,516	1,568	0,0027	0,00029	0,0097	0,96798	27
156/104/104	0,172	2,028	1,291	0,0025	0,00026	0,0086	0,99294	34
156/156/26	0,160	2,422	3,053	0,0029	0,00030	0,0099	0,96649	26
156/156/52	0,189	3,132	4,906	0,0024	0,00025	0,0082	0,99300	31
156/156/104	0,158	2,073	2,857	0,0022	0,00024	0,0079	0,99310	34
182/52/52	0,196	3,191	1,451	0,0029	0,00029	0,0098	0,96679	26
182/52/104	0,187	1,901	1,600	0,0025	0,00025	0,0084	0,99300	35
182/104/52	0,319	3,251	1,969	0,0024	0,00027	0,0089	0,96812	26
182/104/104	0,238	3,376	1,258	0,0023	0,00024	0,0080	0,99302	32
182/156/52	0,235	2,834	1,505	0,0027	0,00028	0,0094	0,97297	27
182/156/104	0,170	2,514	1,359	0,0022	0,00024	0,0079	0,99322	34
208/52/52	0,220	3,813	1,513	0,0028	0,00028	0,0095	0,96800	24
208/52/104	0,327	4,116	1,545	0,0025	0,00025	0,0083	0,99304	30
208/104/26	0,217	3,765	3,205	0,0031	0,00029	0,0101	0,96710	23
208/104/52	0,148	1,919	1,892	0,0024	0,00027	0,0089	0,96828	30
208/104/104	0,122	1,937	1,252	0,0022	0,00024	0,0079	0,99312	35
208/156/26	0,393	5,097	1,420	0,0027	0,00027	0,0092	0,96910	23
208/156/52	0,281	3,283	1,643	0,0027	0,00025	0,0084	0,99307	32
208/156/104	0,176	2,044	1,582	0,0023	0,00024	0,0080	0,99314	35
104/104/52/26	0,365	4,586	2,162	0,0033	0,00030	0,0102	0,96659	21
104/104/104/26	0,196	2,772	1,437	0,0024	0,00025	0,0082	0,99302	34
104/156/52/26	0,259	2,444	1,750	0,0024	0,00027	0,0090	0,96787	28
104/156/104/26	0,213	3,685	1,921	0,0022	0,00024	0,0080	0,99310	32
156/104/52/26	0,323	3,919	1,848	0,0024	0,00027	0,0089	0,96858	25
156/104/104/26	0,431	5,417	3,315	0,0023	0,00024	0,0080	0,99291	27
156/156/52/26	0,215	2,751	2,599	0,0028	0,00028	0,0095	0,96722	25
156/156/104/26	0,304	3,455	1,701	0,0021	0,00024	0,0078	0,99323	31
208/104/52/26	0,214	3,180	2,520	0,0025	0,00027	0,0090	0,97135	26
208/104/104/26	0,279	4,514	1,673	0,0021	0,00024	0,0077	0,99330	31
208/156/52/26	0,333	3,816	1,452	0,0025	0,00026	0,0088	0,96768	25
208/156/104/26	0,206	3,255	1,452	0,0020	0,00024	0,0077	0,99331	32
208/156/104/52	0,185	2,428	1,352	0,0020	0,00023	0,0076	0,99336	34
208/156/104/156	0,335	3,747	2,469	0,0021	0,00023	0,0077	0,99347	30
208/156/208/52	0,400	4,391	1,368	0,0021	0,00023	0,0076	0,99345	29
208/156/208/156	0,199	3,542	1,123	0,0022	0,00023	0,0078	0,99320	33
208/260/104/52	0,302	2,878	1,196	0,0020	0,00023	0,0076	0,99347	32
208/260/104/156	0,264	3,466	1,100	0,0020	0,00023	0,0076	0,99339	32
208/260/208/52	0,304	3,673	2,741	0,0020	0,00023	0,0076	0,99325	30
208/260/208/156	0,170	2,039	1,007	0,0019	0,00024	0,0076	0,99344	35
208/364/104/52	0,146	2,346	1,193	0,0020	0,00023	0,0076	0,99342	34
208/364/104/156	0,222	3,369	1,053	0,0021	0,00023	0,0078	0,99332	32
208/364/208/52	0,215	2,889	2,141	0,0019	0,00023	0,0075	0,99357	33
208/364/208/156	0,195	2,189	0,940	0,0019	0,00023	0,0076	0,99333	35

Tabelle A.9 fortgesetzt.

Neuronen pro Schicht	<i>MAPE</i> in (%)	<i>MAE</i> in (t/h)	<i>MAPE</i> PRL (%)	<i>MAE</i>	<i>MSE</i>	<i>RM- SE</i>	<i>R</i>²	Score
312/156/104/52	0,362	4,943	2,038	0,0020	0,00023	0,0076	0,99320	29
312/156/104/156	0,332	4,793	1,815	0,0022	0,00023	0,0077	0,99346	29
312/156/208/52	0,190	2,207	1,133	0,0020	0,00023	0,0076	0,99355	35
312/156/208/156	0,184	3,219	1,089	0,0019	0,00023	0,0075	0,99347	33
312/260/104/52	0,241	3,010	2,325	0,0020	0,00023	0,0076	0,99351	32
312/260/104/156	0,185	3,023	2,336	0,0019	0,00023	0,0076	0,99335	33
312/260/208/52	0,305	4,010	2,248	0,0018	0,00023	0,0074	0,99347	30
312/260/208/156	0,171	2,201	1,709	0,0021	0,00023	0,0076	0,99355	35
312/364/104/52	0,316	4,735	1,407	0,0020	0,00023	0,0076	0,99346	29
312/364/104/156	0,153	2,244	1,042	0,0019	0,00023	0,0076	0,99348	34
312/364/208/52	0,161	2,375	1,129	0,0020	0,00023	0,0075	0,99354	34
312/364/208/156	0,356	4,707	1,469	0,0020	0,00023	0,0076	0,99344	29
416/156/104/52	0,184	3,456	1,184	0,0020	0,00023	0,0076	0,99341	33
416/156/104/156	0,233	3,877	0,948	0,0020	0,00023	0,0077	0,99339	31
416/156/208/52	0,194	3,141	2,414	0,0018	0,00023	0,0074	0,99354	32
416/156/208/156	0,289	3,282	2,480	0,0020	0,00023	0,0076	0,99322	31
416/260/104/52	0,131	2,250	1,292	0,0018	0,00023	0,0075	0,99333	34
416/260/104/156	0,341	4,256	1,256	0,0019	0,00023	0,0075	0,99352	30
416/260/208/52	0,442	5,196	0,938	0,0020	0,00023	0,0075	0,99337	28
416/260/208/156	0,299	3,944	1,277	0,0019	0,00022	0,0074	0,99364	30
416/364/104/52	0,182	2,084	1,380	0,0018	0,00023	0,0074	0,99356	35
416/364/104/156	0,217	2,895	1,198	0,0020	0,00023	0,0076	0,99348	33
416/364/208/52	0,173	2,689	1,282	0,0020	0,00023	0,0076	0,99360	34
416/364/208/156	0,198	2,943	2,948	0,0019	0,00023	0,0075	0,99333	33
520/416/260/156	0,541	7,775	1,409	0,0021	0,00022	0,0075	0,99357	27
520/416/260/260	0,157	2,550	2,336	0,0018	0,00022	0,0073	0,99358	34
520/416/364/156	0,151	2,458	1,458	0,0019	0,00022	0,0074	0,99357	34
520/416/364/260	0,289	3,824	1,101	0,0018	0,00023	0,0074	0,99352	31
520/520/260/156	0,262	3,827	2,393	0,0022	0,00023	0,0077	0,99369	31
520/520/260/260	0,204	2,620	0,961	0,0019	0,00022	0,0073	0,99380	34
520/520/364/156	0,525	8,166	1,155	0,0022	0,00023	0,0077	0,99340	27
520/520/364/260	0,233	3,580	1,095	0,0023	0,00022	0,0077	0,99368	32
520/624/260/156	0,249	3,442	1,502	0,0019	0,00023	0,0074	0,99364	32
520/624/260/260	0,362	4,748	0,970	0,0020	0,00023	0,0075	0,99363	29
520/624/364/156	0,329	5,842	1,138	0,0023	0,00022	0,0077	0,99368	29
520/624/364/260	0,219	2,765	1,287	0,0019	0,00022	0,0074	0,99354	33
624/416/260/156	0,316	3,687	1,036	0,0019	0,00022	0,0073	0,99371	31
624/416/260/260	0,325	3,768	0,955	0,0021	0,00023	0,0076	0,99340	31
624/416/364/156	0,255	3,666	0,856	0,0019	0,00022	0,0074	0,99376	32
624/416/364/260	0,215	3,815	2,171	0,0019	0,00023	0,0074	0,99366	31
624/520/260/156	0,193	2,589	2,347	0,0018	0,00023	0,0074	0,99348	34
624/520/260/260	0,244	3,741	2,268	0,0019	0,00022	0,0074	0,99387	32
624/520/364/156	0,176	2,146	0,909	0,0017	0,00022	0,0071	0,99383	35
624/520/364/260	0,641	9,202	1,358	0,0020	0,00022	0,0074	0,99376	27
624/624/260/156	0,368	5,179	0,932	0,0019	0,00022	0,0074	0,99374	29
624/624/260/260	0,280	4,570	0,983	0,0020	0,00022	0,0073	0,99388	30

Tabelle A.9 fortgesetzt.

Neuronen pro Schicht	<i>MAPE</i> \hat{m} (%)	<i>MAE</i> \hat{m} (t/h)	<i>MAPE</i> PRL (%)	<i>MAE</i>	<i>MSE</i>	<i>RM- SE</i>	R^2	Score
624/624/364/156	0,225	3,069	1,409	0,0019	0,00022	0,0072	0,99378	32
624/624/364/260	0,140	2,146	1,114	0,0018	0,00022	0,0073	0,99357	35
728/416/260/156	0,234	3,017	1,044	0,0018	0,00020	0,0069	0,99403	32
728/416/260/260	0,303	4,381	1,371	0,0022	0,00022	0,0075	0,99382	30
728/416/364/156	0,179	2,369	1,260	0,0019	0,00021	0,0073	0,99370	34
728/416/364/260	0,349	4,161	2,870	0,0021	0,00022	0,0076	0,99351	29
728/520/260/156	0,557	9,023	2,954	0,0021	0,00020	0,0072	0,99387	26
728/520/260/260	0,204	2,887	2,308	0,0020	0,00022	0,0075	0,99370	33
728/520/364/156	0,422	6,553	1,212	0,0021	0,00020	0,0071	0,99407	28
728/520/364/260	0,238	2,689	0,995	0,0019	0,00022	0,0073	0,99384	33
728/624/260/156	0,222	3,460	1,469	0,0022	0,00020	0,0073	0,99397	32
728/624/260/260	0,325	5,721	1,086	0,0019	0,00020	0,0070	0,99398	29
728/624/364/156	0,430	5,319	2,278	0,0018	0,00021	0,0072	0,99393	28
728/624/364/260	0,176	2,006	0,859	0,0018	0,00022	0,0073	0,99367	35

Tabelle A.10.: Bewertungsscore neuronaler Netze für Fernwärmenetz 2.

	<i>MAPE</i> \hat{m} (%)	<i>MAE</i> \hat{m} (t/h)	<i>MAPE</i> PRL (%)	<i>MAE</i>	<i>MSE</i>	<i>RMSE</i>	R^2
Minimum	0,117	1,438	0,856	0,00172	0,00020	0,00695	0,94174
Maximum	0,989	9,265	16,395	0,01202	0,00095	0,02205	0,99407
Punkte							
5	< 0,204	< 2,220	< 0,291	< 0,00275	< 0,00028	< 0,00846	> 0,98884
4	< 0,291	< 3,964	< 0,291	< 0,00378	< 0,00035	< 0,00997	> 0,98361
3	< 0,378	< 5,518	< 0,291	< 0,00481	< 0,00043	< 0,01148	< 0,97837
2	< 0,466	< 7,072	< 0,291	< 0,00584	< 0,00050	< 0,01299	> 0,97314
1	Rest	Rest	Rest	Rest	Rest	Rest	Rest

Die folgende Tabelle A.11 beschreibt alle Ausgangsgrößen der trainierten neuronalen Netze. Zudem wird angegeben, ob die Ausgangsgröße in die Berechnung der relativen Modellgenauigkeit in Kapitel 6.2.2 einfließt. Einige Ausgänge werden in der späteren Optimierung (OPT) nicht mehr verwendet und sind daher nicht relevant. Dazu zählen zum Beispiel Pumpenleistungen, welche für die Optimierung der Einspeiseleistung und Vorlauftemperaturen der Wärmeerzeugungsanlagen nicht notwendig sind. Diese würden nur für weitergehende Optimierungen, zum Beispiel der Gesamtleistung des Systems verwendet werden und müssten für diesen Anwendungsfall nachgebessert werden. Andere Modellausgänge verursachen nur hohe relative Abweichungen. Zum Beispiel das Heizwerk und die Druckerhöhungsanlage sind in den Simulationen häufig ausgestellt, weshalb die Sollwerte nach *SIR 3S*[®] 0 entsprechen, das neuronale Netz aber kleine absolute Werte ungleich 0 vorhersagt. Somit führen hier absolut kleine Werte zu hohen relativen Abweichungen. Eine Auswertung dieser Größen für den durchschnittlichen relativen Fehler ist nicht sinnvoll.

Tabelle A.11.: Ausgangsgrößen der neuronalen Netze für Fernwärmenetz 2 und Verwendung für die Berechnung der relativen Modellgenauigkeit.

Index	Modellausgang	Für Auswertung der Modellgenauigkeit verwendet?	Warum nicht?
0	Wärmeverbrauch	Nein	Nicht für OPT nötig.
1	P HKW	Ja	
2	\dot{m} gesamt	Ja	
3	\dot{m} HKW	Ja	
4	\dot{m} HW	Nein	Werte oft nahe 0.
5	\dot{m} DEA	Nein	Nicht für OPT nötig.
6	v HW	Nein	Werte oft nahe 0.
7	v DEA	Nein	Nicht für OPT nötig.
8	v HKW	Ja	
9	Min. Δp	Ja	
10	Max. p_{RL}	Ja	
11	Min. p_{RL}	Ja	
12	Max. p_{VL}	Ja	
13	Min. p_{VL}	Ja	
14	Min. T_{VL}	Nein	Mittlerweile behobene, fehlerhafte Ausgabe des Interfaces zu <i>SIR 3S</i> [®] .
15	Wärmeverlust	Nein	Nicht für OPT nötig.
16	P_{Pumpe} VL HKW	Nein	Nicht für OPT nötig.
17	P_{Pumpe} RL HKW	Nein	Nicht für OPT nötig.
18	P_{Pumpe} RL HW	Nein	Nicht für OPT nötig.
19	P_{Pumpe} RL DEA	Nein	Nicht für OPT nötig.
20	n_{Pumpe} VL HKW	Ja	
21	n_{Pumpe} RL HKW	Ja	
22	n_{Pumpe} RL HW	Nein	Werte oft nahe 0.
23	n_{Pumpe} RL DEA	Nein	Werte oft nahe 0.
24	p_{VL} HKW	Nein	Nicht für OPT nötig.
25	p_{RL} HKW	Nein	Nicht für OPT nötig.

In Kapitel 6.2.2 wird das neuronale Netz mit vier versteckten Schichten und jeweils 208, 260, 208 und 156 Neuronen für Fernwärmenetz 2 ausgewählt. Die Tabellen A.12 bis A.14 zeigen die Auswertung des Netzes mit und ohne Rücklauftemperatureingang für jeweils zehn Simulationsvariationen basierend auf den drei Beispieltagen ausgewertet. Dargestellt sind die maximalen absoluten und relativen Abweichungen sowie der MAPE für alle Ausgangsgrößen. Relative Abweichungen größer 3% sind rot markiert. Einige der Abweichungen sind grau markiert, falls die hohen relativen Abweichungen lediglich durch Ausgangswerte nahe 0 entstehen. Bei dem Ausgang minimale Vorlauftemperatur sind hohe Abweichungen ebenfalls grau markiert, da diese durch inzwischen behobene Berechnungsfehler in *SIR 3S*[®] entstanden sind.

Tabelle A.12.: Maximale absolute und relative Abweichungen der neuronalen Netze zu $SIR\ 3S^{\circledast}$ für das aggregierte Netz 2 und für zehn Simulationstage basierend auf Beispieltag 1 (Februar).

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	$MAPE$ mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	$MAPE$ ohne T_{RL} (%)
Wärmeverbrauch (MW)	0,076	0,048	0,012	0,188	0,125	0,037
P HKW (MW)	0,095	0,071	0,014	0,116	0,100	0,037
\dot{m} gesamt (t/h)	1,579	0,083	0,015	80,846	3,299	1,092
\dot{m} HKW (t/h)	0,934	0,058	0,010	61,836	3,374	1,113
\dot{m} HW (t/h)	0,636	0,215	0,029	28,080	3,259	1,033
\dot{m} DEA (t/h)	1,192	0,164	0,021	32,312	3,258	1,116
v HW (m/s)	0,001	0,225	0,032	0,022	3,298	1,027
v DEA (m/s)	0,001	0,157	0,022	0,033	3,347	1,149
v HKW (m/s)	0,001	0,060	0,014	0,064	3,379	1,115
Min. Δp (bar)	0,021	0,739	0,139	0,110	4,091	0,702
Max. p_{RL} (bar)	0,021	0,485	0,044	0,054	1,194	0,056
Min. p_{RL} (bar)	0,014	0,532	0,116	0,104	4,775	0,831
Max. p_{VL} (bar)	0,013	0,116	0,024	0,160	1,432	0,399
Min. p_{VL} (bar)	0,011	0,178	0,038	0,025	0,406	0,046
Min. T_{VL} ($^{\circ}C$)	71,926	163,574	0,585	71,826	163,347	0,616
Wärmeverlust (MW)	0,029	0,367	0,019	0,118	1,419	0,456
P_{Pumpe} VL HKW (kW)	1,049	0,446	0,082	23,252	6,983	2,098
P_{Pumpe} RL HKW (kW)	0,863	0,444	0,074	19,953	7,667	1,998
P_{Pumpe} RL HW (kW)	0,696	0,926	0,166	2,741	2,452	0,515
P_{Pumpe} RL DEA (kW)	0,016	inf	inf	0,025	inf	inf
n_{Pumpe} VL HKW (min^{-1})	175,417	17,450	0,094	167,989	16,592	0,729
n_{Pumpe} RL HKW (min^{-1})	1,773	0,190	0,029	25,299	2,387	0,603
n_{Pumpe} RL HW (min^{-1})	3,162	0,335	0,056	8,012	0,845	0,123
n_{Pumpe} RL DEA (min^{-1})	0,065	inf	inf	0,225	inf	inf
p_{VL} HKW (bar)	0,013	0,118	0,024	0,162	1,476	0,409
p_{RL} HKW (bar)	0,012	0,405	0,070	0,158	5,743	0,997

Tabelle A.13.: Maximale absolute und relative Abweichungen der neuronalen Netze zu *SIR 3S*[®] für das aggregierte Netz 2 und für zehn Simulationstage basierend auf Beispieltag 2 (Mai).

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	<i>MAPE</i> mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	<i>MAPE</i> ohne T_{RL} (%)
Wärmeverbrauch (MW)	0,283	0,461	0,150	0,485	0,792	0,163
P HKW (MW)	0,120	0,199	0,084	0,544	1,011	0,398
\dot{m} gesamt (t/h)	2,096	0,182	0,091	255,078	14,486	2,880
\dot{m} HKW (t/h)	0,786	0,069	0,013	239,092	13,798	2,965
\dot{m} HW (t/h)	1,357	8,500	4,406	13,341	72,183	17,857
\dot{m} DEA (t/h)	2,185	0,283	0,063	140,331	13,547	3,341
v HW (m/s)	0,001	7,904	4,803	0,010	74,680	18,023
v DEA (m/s)	0,002	0,268	0,068	0,142	13,779	3,380
v HKW (m/s)	0,001	0,101	0,022	0,243	13,818	2,969
Min. Δp (bar)	0,054	1,546	0,161	0,169	5,328	1,661
Max. p_{RL} (bar)	0,010	0,228	0,019	0,020	0,450	0,073
Min. p_{RL} (bar)	0,017	0,726	0,113	0,328	15,688	1,867
Max. p_{VL} (bar)	0,046	0,478	0,041	0,562	5,049	0,693
Min. p_{VL} (bar)	0,010	0,173	0,031	0,011	0,178	0,038
Min. T_{VL} (°C)	75,963	721,420	3,529	76,632	726,880	3,703
Wärmeverlust (MW)	0,011	0,157	0,016	0,304	4,822	1,045
P_{Pumpe} VL HKW (kW)	1,487	1,226	0,118	75,250	27,349	5,210
P_{Pumpe} RL HKW (kW)	1,405	0,999	0,069	65,890	28,332	4,417
P_{Pumpe} RL HW (kW)	1,031	14,244	8,008	5,908	77,287	17,652
P_{Pumpe} RL DEA (kW)	0,019	inf	inf	0,070	inf	inf
n_{Pumpe} VL HKW (min^{-1})	190,960	16,930	0,133	149,612	12,193	1,594
n_{Pumpe} RL HKW (min^{-1})	142,634	16,064	0,350	146,922	16,459	1,695
n_{Pumpe} RL HW (min^{-1})	109,962	11,930	4,552	40,868	4,200	1,461
n_{Pumpe} RL DEA (min^{-1})	0,172	inf	inf	0,668	inf	inf
p_{VL} HKW (bar)	0,037	0,404	0,037	0,570	5,236	0,745
p_{RL} HKW (bar)	0,011	0,320	0,029	0,548	19,395	1,485

Tabelle A.14.: Maximale absolute und relative Abweichungen der neuronalen Netze zu *SIR 3S*[®] für das aggregierte Netz 2 und für zehn Simulationstage basierend auf Beispieltag 3 (Juli).

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	<i>MAPE</i> mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	<i>MAPE</i> ohne T_{RL} (%)
Wärmeverbrauch (MW)	1,277	40,815	11,180	1,331	41,346	13,339
P HKW (MW)	0,110	1,218	0,171	0,112	1,227	0,313
\dot{m} gesamt (t/h)	1,678	0,665	0,240	26,206	8,302	1,937
\dot{m} HKW (t/h)	3,286	1,982	0,216	23,055	7,852	1,787
\dot{m} HW (t/h)	2,789	13,472	2,637	4,918	23,032	5,543
\dot{m} DEA (t/h)	38,919	49,184	15,043	39,338	58,115	16,437
v HW (m/s)	0,002	14,365	2,971	0,004	24,100	5,843
v DEA (m/s)	0,039	49,130	15,004	0,039	58,012	16,425
v HKW (m/s)	0,004	2,082	0,231	0,023	7,839	1,784
Min. Δp (bar)	0,022	0,536	0,026	0,022	0,521	0,054
Max. p_{RL} (bar)	0,001	0,029	0,008	0,002	0,037	0,010
Min. p_{RL} (bar)	0,094	5,173	0,545	0,098	5,391	0,647
Max. p_{VL} (bar)	0,005	0,057	0,019	0,013	0,143	0,055
Min. p_{VL} (bar)	0,010	0,177	0,043	0,012	0,201	0,052
Min. T_{VL} (°C)	46,284	462,721	34,093	47,296	472,942	35,236
Wärmeverlust (MW)	0,250	4,196	1,779	0,258	4,445	1,050
P_{Pumpe} VL HKW (kW)	0,691	2,385	0,534	1,988	5,818	1,151
P_{Pumpe} RL HKW (kW)	0,533	1,246	0,208	1,524	3,752	1,097
P_{Pumpe} RL HW (kW)	1,741	15,189	3,420	1,787	15,585	3,103
P_{Pumpe} RL DEA (kW)	0,019	inf	inf	0,038	inf	inf
n_{Pumpe} VL HKW (min^{-1})	1,070	0,140	0,039	2,649	0,347	0,071
n_{Pumpe} RL HKW (min^{-1})	0,327	0,041	0,010	2,156	0,268	0,071
n_{Pumpe} RL HW (min^{-1})	24,941	2,317	0,388	15,754	1,461	0,351
n_{Pumpe} RL DEA (min^{-1})	0,074	inf	inf	0,756	inf	inf
p_{VL} HKW (bar)	0,006	0,067	0,021	0,013	0,156	0,051
p_{RL} HKW (bar)	0,002	0,037	0,009	0,006	0,150	0,053

A.1.4. Modellvalidierung Netz 3

Hier werden zusätzliche Informationen zur Auswahl des neuronalen Netzes für Fernwärmenetz 3 beschrieben. Die trainierten Modelle mit der Rücklaufemperatur als Eingang haben 17 Eingänge und 43 Ausgänge. Folgende Eingänge werden für eine Simulation benötigt:

- Gesamte Wärmeeinspeisung
- Relativer Wärmeverbrauch in Teilgebiet 2
- Leistungen von HW 1, HW 3 und HW 2
- Vorlaufemperaturen von HKW, HW 1, HW 3 und HW 2
- *Booleans* für die fünf Betriebsarten der DEA sowie ein *Boolean* für den Fall der fixen Vorgabe der Minimaldrehzahl
- Rücklaufemperaturen von HKW und HW 3

Die Ausgangsgrößen sind in Tabelle A.17 gegeben. Für das Training neuronaler Netze wird die Anzahl der Neuronen jeweils als Vielfaches der Anzahl der Ausgänge gewählt, die Anzahl der Schichten zwischen 1 und 4 variiert. Das Training wird mit *PyTorch* und den folgenden Parametern durchgeführt:

- 60%/20%/20% Unterteilung zwischen Trainings-, Validierungs- und Testdaten und Normierung der Ein- und Ausgangsdaten zwischen 0 und 1
- Optimierer: *Adam*
- Loss-Funktion: MAE
- Aktivierungsfunktion: *Swish*
- Epochen: 100

In Tabelle A.15 ist die Auswertung der trainierten Modelle dargestellt. Für die Ausgangsgrößen minimaler Differenzdruck im Teilgebiet 2 (Δp), den maximalen Rücklaufdruck in Teilgebiet 1 (p_{RL}) und den Massenstrom des HKW \dot{m} sind der MAPE oder MAE angegeben. Diese Größen variieren zwischen den Modellen etwas stärker als die restlichen skalierten Werte über alle Ausgangsgrößen MAE, MSE, RMSE und R^2 . Zusätzlich ist ein Bewertungskriterium *Score* in der Tabelle angegeben. Hier werden für jedes Modell 1 bis 5 Punkte entsprechend Tabelle A.16 vergeben. Die nach dieser Bewertung ausgewählten Modelle sind in Tabelle A.15 grün markiert. Es wird jeweils das beste neuronale Netz mit einer, zwei, drei und vier Schichten gewählt. Bei mehreren Netzen mit gleicher Bewertung wird das Modell mit der geringeren Anzahl an Neuronen selektiert. Für drei und vier Schichten sind mehrere Modelle mit einer hohen Bewertung ausgewählt.

Tabelle A.15.: Bewertung neuronaler Netze mit Testdaten für Fernwärmenetz 3 nach 100 Epochen Training.

Neuronen pro Schicht	MA-PE Δp (%)	MAE PRL (bar)	MAE \dot{m} (t/h)	MAE	MSE	RM-SE	R ²	Score
43	7,957	0,042	16,178	0,0087	0,00091	0,0202	0,92615	7
86	6,702	0,031	15,349	0,0078	0,00070	0,0175	0,93943	7
129	6,194	0,028	14,284	0,0070	0,00053	0,0159	0,95371	8
172	6,034	0,032	14,139	0,0074	0,00058	0,0163	0,94830	7
215	5,682	0,028	14,221	0,0072	0,00054	0,0159	0,94444	7

Tabelle A.15 fortgesetzt.

Neuronen pro Schicht	MA- PE Δp (%)	MAE PRL (bar)	MAE m (t/h)	MAE	MSE	RM- SE	R ²	Score
258	6,393	0,026	16,095	0,0072	0,00056	0,0160	0,95277	7
301	5,626	0,026	14,813	0,0073	0,00055	0,0161	0,94915	7
344	6,350	0,026	16,315	0,0071	0,00055	0,0158	0,94687	7
43/86	4,698	0,019	9,147	0,0047	0,00027	0,0117	0,96315	12
43/172	4,517	0,018	12,532	0,0048	0,00030	0,0117	0,96946	12
43/258	4,399	0,017	8,524	0,0046	0,00025	0,0111	0,97346	15
86/86	3,724	0,015	6,322	0,0040	0,00022	0,0103	0,96687	17
86/172	4,358	0,015	10,643	0,0041	0,00020	0,0100	0,97184	16
86/258	3,676	0,024	12,094	0,0041	0,00021	0,0100	0,97462	14
129/86	3,878	0,014	6,707	0,0038	0,00018	0,0095	0,97728	20
129/172	3,419	0,018	7,835	0,0038	0,00018	0,0094	0,97783	19
129/258	3,143	0,016	6,095	0,0036	0,00019	0,0095	0,97169	21
172/86	3,492	0,016	7,017	0,0036	0,00020	0,0098	0,97189	19
172/172	2,991	0,015	5,649	0,0032	0,00017	0,0088	0,98131	24
172/258	2,837	0,013	8,759	0,0033	0,00015	0,0085	0,97556	22
215/86	3,148	0,016	6,822	0,0035	0,00017	0,0090	0,97035	21
215/172	2,748	0,013	10,316	0,0035	0,00015	0,0087	0,97457	21
215/258	2,681	0,013	6,862	0,0031	0,00014	0,0083	0,97601	23
258/86	3,131	0,016	6,108	0,0034	0,00019	0,0092	0,97683	21
258/172	2,721	0,014	5,877	0,0031	0,00014	0,0083	0,98186	25
258/258	3,059	0,018	6,471	0,0032	0,00014	0,0082	0,98145	22
301/86	3,308	0,015	5,646	0,0036	0,00019	0,0094	0,98022	23
301/172	2,962	0,015	9,163	0,0033	0,00016	0,0088	0,97340	21
301/258	2,757	0,019	16,986	0,0034	0,00014	0,0084	0,96967	19
344/86	2,919	0,015	7,704	0,0033	0,00017	0,0090	0,98061	22
344/172	2,582	0,015	5,867	0,0031	0,00014	0,0083	0,97283	24
344/258	2,404	0,014	7,945	0,0030	0,00014	0,0082	0,97605	22
43/86/86	2,982	0,017	6,010	0,0032	0,00017	0,0089	0,98076	23
43/86/172	3,366	0,013	8,412	0,0031	0,00014	0,0082	0,98029	22
43/172/86	3,047	0,014	7,123	0,0029	0,00013	0,0079	0,98207	24
43/172/172	2,492	0,022	14,674	0,0030	0,00012	0,0078	0,98076	21
43/258/86	2,467	0,014	10,842	0,0031	0,00014	0,0081	0,97289	21
43/258/172	2,839	0,012	5,607	0,0027	0,00012	0,0075	0,97866	27
86/86/86	2,353	0,014	6,859	0,0029	0,00013	0,0078	0,97377	25
86/86/172	2,801	0,012	8,728	0,0029	0,00014	0,0080	0,97678	23
86/172/86	2,406	0,012	7,458	0,0026	0,00012	0,0072	0,98070	27
86/172/172	2,126	0,014	6,818	0,0026	0,00012	0,0075	0,97641	26
86/258/86	1,997	0,014	5,913	0,0025	0,00012	0,0075	0,97511	27
86/258/172	2,258	0,015	7,813	0,0027	0,00011	0,0069	0,97841	24
129/86/86	2,513	0,012	7,885	0,0026	0,00013	0,0078	0,98190	26
129/86/172	2,420	0,011	8,089	0,0026	0,00013	0,0076	0,97571	25
129/172/86	2,215	0,025	12,631	0,0029	0,00014	0,0080	0,98041	20
129/172/172	2,371	0,011	4,972	0,0027	0,00011	0,0070	0,97890	28
129/258/86	2,530	0,013	6,305	0,0024	0,00012	0,0073	0,97989	26

Tabelle A.15 fortgesetzt.

Neuronen pro Schicht	MA- PE Δp (%)	MAE PRL (bar)	MAE \dot{m} (t/h)	MAE	MSE	RM- SE	R ²	Score
129/258/172	1,998	0,012	6,493	0,0025	0,00012	0,0073	0,98285	27
172/86/86	2,441	0,017	8,842	0,0028	0,00013	0,0078	0,98235	24
172/86/172	2,225	0,010	5,638	0,0024	0,00013	0,0075	0,97544	27
172/86/43	2,810	0,015	9,653	0,0028	0,00015	0,0082	0,98150	23
172/172/43	2,036	0,013	6,617	0,0025	0,00013	0,0077	0,97696	26
172/172/86	1,789	0,011	6,975	0,0023	0,00011	0,0071	0,98466	28
172/172/172	2,005	0,013	7,117	0,0024	0,00011	0,0069	0,98550	27
172/258/43	2,002	0,015	4,950	0,0026	0,00014	0,0079	0,97528	27
172/258/86	2,048	0,010	6,080	0,0022	0,00010	0,0065	0,97654	28
172/258/172	1,850	0,010	7,188	0,0025	0,00011	0,0072	0,97569	27
215/86/86	2,294	0,013	6,320	0,0026	0,00012	0,0073	0,98303	26
215/86/172	1,888	0,011	6,271	0,0026	0,00013	0,0077	0,98258	28
215/172/86	1,994	0,010	5,952	0,0024	0,00012	0,0072	0,97973	29
215/172/172	2,060	0,009	5,238	0,0023	0,00011	0,0066	0,98174	29
215/258/86	2,159	0,010	5,989	0,0023	0,00011	0,0065	0,98143	28
215/258/172	1,902	0,009	6,697	0,0023	0,00011	0,0070	0,97617	27
258/86/43	2,221	0,013	6,725	0,0027	0,00014	0,0079	0,98167	26
258/86/86	2,045	0,012	6,803	0,0023	0,00011	0,0067	0,98229	28
258/86/172	2,092	0,010	5,566	0,0024	0,00013	0,0076	0,97858	29
258/172/43	2,168	0,012	6,851	0,0024	0,00011	0,0070	0,98296	26
258/172/86	2,746	0,010	6,311	0,0025	0,00012	0,0074	0,97977	27
258/172/172	1,810	0,011	5,648	0,0023	0,00012	0,0072	0,97962	29
258/258/43	2,067	0,011	7,105	0,0027	0,00012	0,0072	0,97833	27
258/258/86	1,975	0,011	6,317	0,0026	0,00010	0,0067	0,98701	29
258/258/172	1,754	0,013	7,998	0,0023	0,00011	0,0067	0,97831	25
301/86/86	2,220	0,013	7,324	0,0025	0,00013	0,0074	0,98134	26
301/86/172	2,014	0,008	7,013	0,0024	0,00013	0,0075	0,97438	27
301/172/86	1,927	0,010	5,476	0,0022	0,00010	0,0065	0,98234	29
301/172/172	1,908	0,018	9,080	0,0024	0,00010	0,0064	0,98685	26
301/258/86	1,672	0,009	5,583	0,0022	0,00010	0,0063	0,98410	29
301/258/172	1,653	0,011	6,293	0,0022	0,00010	0,0062	0,98207	29
344/86/43	2,323	0,015	6,078	0,0026	0,00013	0,0078	0,97594	26
344/86/86	1,939	0,012	7,058	0,0025	0,00013	0,0075	0,97392	27
344/86/172	2,463	0,011	5,899	0,0025	0,00013	0,0075	0,97820	27
344/172/43	2,168	0,010	4,870	0,0023	0,00011	0,0071	0,97594	27
344/172/86	1,896	0,009	5,560	0,0022	0,00011	0,0070	0,97423	28
344/172/172	1,698	0,010	6,364	0,0022	0,00010	0,0063	0,97948	28
344/258/43	1,626	0,012	7,214	0,0023	0,00010	0,0065	0,98109	27
344/258/86	1,648	0,010	8,024	0,0021	0,00009	0,0061	0,98227	29
344/258/172	1,698	0,009	6,381	0,0025	0,00011	0,0066	0,97927	28
172/86/86/43	2,161	0,014	6,561	0,0024	0,00011	0,0071	0,98486	26
172/86/172/43	1,776	0,008	5,940	0,0023	0,00011	0,0070	0,97644	29
172/172/86/43	2,126	0,011	9,231	0,0023	0,00011	0,0071	0,97555	25
172/172/172/43	2,003	0,014	8,743	0,0024	0,00010	0,0065	0,98122	26

Tabelle A.15 fortgesetzt.

Neuronen pro Schicht	MA- PE Δp (%)	MAE PRL (bar)	MAE m (t/h)	MAE	MSE	RM- SE	R ²	Score
172/258/86/43	1,698	0,013	5,929	0,0023	0,00011	0,0069	0,98145	28
172/258/172/43	1,587	0,012	9,959	0,0023	0,00010	0,0068	0,98208	26
258/86/86/43	1,865	0,010	6,641	0,0024	0,00011	0,0073	0,97319	27
258/86/172/43	1,459	0,012	10,270	0,0024	0,00010	0,0064	0,97984	26
258/172/86/43	1,717	0,012	5,838	0,0021	0,00011	0,0068	0,98266	30
258/172/172/43	1,541	0,009	6,346	0,0021	0,00009	0,0061	0,98424	30
258/258/86/43	1,678	0,009	6,089	0,0023	0,00010	0,0062	0,98329	30
258/258/172/43	1,391	0,009	4,453	0,0019	0,00009	0,0059	0,98166	33
344/86/86/43	1,753	0,015	7,800	0,0022	0,00010	0,0063	0,97646	25
344/86/172/43	1,539	0,010	4,948	0,0021	0,00010	0,0063	0,97862	30
344/172/86/43	1,564	0,010	10,051	0,0022	0,00010	0,0062	0,98213	27
344/172/172/43	1,185	0,010	9,151	0,0021	0,00009	0,0060	0,98281	30
344/258/86/43	1,474	0,013	12,154	0,0023	0,00010	0,0065	0,97761	24
344/258/172/43	1,189	0,011	6,615	0,0020	0,00009	0,0059	0,98692	32
344/258/86/172	1,737	0,011	7,645	0,0022	0,00010	0,0063	0,98301	29
344/258/86/344	1,585	0,009	6,051	0,0022	0,00009	0,0061	0,98944	31
344/258/86/516	1,620	0,012	6,918	0,0022	0,00010	0,0062	0,98902	29
344/258/344/172	1,559	0,009	6,363	0,0022	0,00009	0,0061	0,98213	29
344/258/344/344	1,670	0,015	8,944	0,0026	0,00010	0,0064	0,98915	27
344/258/344/516	1,583	0,011	8,056	0,0021	0,00009	0,0061	0,98258	29
344/516/86/172	1,862	0,012	7,131	0,0024	0,00010	0,0064	0,98136	28
344/516/86/344	1,802	0,012	5,198	0,0023	0,00010	0,0063	0,98672	30
344/516/86/516	1,652	0,011	6,676	0,0023	0,00010	0,0063	0,98264	28
344/516/344/172	1,374	0,008	7,944	0,0020	0,00009	0,0059	0,98329	31
344/516/344/344	1,505	0,010	5,594	0,0022	0,00009	0,0060	0,98359	30
344/516/344/516	1,380	0,008	5,070	0,0020	0,00010	0,0061	0,98272	33
344/774/86/172	1,608	0,013	7,151	0,0023	0,00009	0,0061	0,98774	29
344/774/86/344	1,638	0,017	10,746	0,0025	0,00010	0,0063	0,98314	24
344/774/86/516	1,558	0,022	18,266	0,0025	0,00010	0,0065	0,98882	24
344/774/344/172	1,408	0,008	5,244	0,0020	0,00009	0,0059	0,98325	33
344/774/344/344	1,434	0,015	11,047	0,0024	0,00009	0,0063	0,97772	24
344/774/344/516	1,677	0,013	8,162	0,0023	0,00010	0,0067	0,98405	26
688/258/86/172	1,121	0,011	10,282	0,0020	0,00009	0,0059	0,99102	30
688/258/86/344	1,303	0,010	6,983	0,0021	0,00009	0,0061	0,98030	31
688/258/86/516	1,328	0,012	8,815	0,0021	0,00009	0,0061	0,98289	30
688/258/344/172	1,591	0,018	13,853	0,0026	0,00009	0,0062	0,98125	25
688/258/344/344	1,280	0,010	8,041	0,0020	0,00009	0,0059	0,97734	29
688/258/344/516	1,436	0,012	11,267	0,0021	0,00009	0,0061	0,98066	28
688/516/86/172	1,375	0,011	8,048	0,0021	0,00009	0,0059	0,98869	31
688/516/86/344	1,381	0,011	6,691	0,0024	0,00009	0,0062	0,98998	31
688/516/86/516	1,287	0,008	7,727	0,0022	0,00009	0,0061	0,98723	30
688/516/344/172	1,216	0,009	6,415	0,0020	0,00009	0,0058	0,98357	31
688/516/344/344	1,635	0,011	7,687	0,0021	0,00009	0,0059	0,97866	30
688/516/344/516	1,296	0,015	6,771	0,0021	0,00009	0,0059	0,98117	30

Tabelle A.15 fortgesetzt.

Neuronen pro Schicht	MA- PE Δp (%)	MAE PRL (bar)	MAE \dot{m} (t/h)	MAE	MSE	RM- SE	R ²	Score
688/774/86/172	1,386	0,011	6,834	0,0022	0,00009	0,0061	0,99034	31
688/774/86/344	1,470	0,008	6,093	0,0019	0,00009	0,0058	0,98024	32
688/774/86/516	1,612	0,010	5,834	0,0020	0,00009	0,0059	0,98788	32
688/774/344/172	1,296	0,010	9,030	0,0021	0,00009	0,0058	0,99086	31
688/774/344/344	1,409	0,008	7,155	0,0019	0,00009	0,0057	0,98254	32
688/774/344/516	1,422	0,011	11,668	0,0022	0,00009	0,0060	0,97954	28
1032/258/86/172	1,490	0,015	12,308	0,0023	0,00009	0,0060	0,98080	26
1032/258/86/344	1,483	0,012	10,527	0,0020	0,00009	0,0059	0,98536	27
1032/258/86/516	1,468	0,010	5,411	0,0023	0,00009	0,0062	0,98029	30
1032/258/344/172	1,045	0,012	7,731	0,0022	0,00009	0,0058	0,98610	29
1032/258/344/344	1,223	0,010	6,820	0,0021	0,00009	0,0059	0,99100	32
1032/258/344/516	1,448	0,009	6,128	0,0020	0,00009	0,0058	0,98990	32
1032/516/86/172	1,145	0,012	12,832	0,0021	0,00009	0,0059	0,97949	28
1032/516/86/344	1,536	0,012	9,908	0,0023	0,00009	0,0060	0,98151	26
1032/516/86/516	1,559	0,017	17,802	0,0024	0,00009	0,0063	0,99004	25
1032/516/344/172	1,062	0,013	8,364	0,0020	0,00009	0,0060	0,98481	29
1032/516/344/344	1,292	0,009	6,275	0,0019	0,00009	0,0058	0,98144	31
1032/516/344/516	1,456	0,008	5,827	0,0018	0,00009	0,0056	0,98772	33
1032/774/86/172	1,615	0,010	10,243	0,0022	0,00009	0,0060	0,98267	27
1032/774/86/344	1,224	0,009	4,724	0,0019	0,00009	0,0058	0,98177	32
1032/774/86/516	1,831	0,010	5,559	0,0019	0,00010	0,0065	0,97760	29
1032/774/344/172	1,573	0,009	6,005	0,0022	0,00009	0,0058	0,98494	30
1032/774/344/344	1,272	0,014	9,959	0,0022	0,00009	0,0059	0,98464	27
1032/774/344/516	1,231	0,008	6,814	0,0021	0,00009	0,0058	0,98237	32
1032/774/516/516	1,493	0,009	7,689	0,0021	0,00009	0,0060	0,99029	31
1032/774/516/774	1,242	0,012	13,348	0,0023	0,00009	0,0061	0,98271	28
1032/774/774/516	1,181	0,011	15,012	0,0022	0,00009	0,0060	0,99059	29
1032/774/774/774	1,432	0,007	6,710	0,0023	0,00009	0,0061	0,98450	30
1032/1032/516/516	1,449	0,013	8,827	0,0024	0,00009	0,0061	0,98282	27
1032/1032/516/774	1,324	0,008	5,807	0,0020	0,00009	0,0059	0,98381	32
1032/1032/774/516	1,696	0,008	7,486	0,0022	0,00009	0,0060	0,97863	30
1032/1032/774/774	1,042	0,010	9,300	0,0022	0,00007	0,0054	0,98904	29
1290/774/516/516	1,170	0,009	7,543	0,0021	0,00009	0,0059	0,98317	31
1290/774/516/774	1,305	0,014	8,168	0,0022	0,00008	0,0058	0,98252	28
1290/774/774/516	1,316	0,013	7,510	0,0022	0,00009	0,0060	0,98319	29
1290/774/774/774	1,649	0,012	7,400	0,0021	0,00009	0,0059	0,98287	30
1290/1032/516/516	1,225	0,009	5,895	0,0020	0,00009	0,0057	0,98359	32
1290/1032/516/774	1,485	0,009	4,506	0,0019	0,00007	0,0054	0,98644	33
1290/1032/774/516	1,242	0,008	7,798	0,0022	0,00009	0,0059	0,98785	31
1290/1032/774/774	1,223	0,008	7,639	0,0020	0,00009	0,0057	0,98721	33
1548/774/516/516	1,192	0,013	8,266	0,0020	0,00007	0,0054	0,99020	30
1548/774/516/774	1,221	0,005	5,636	0,0018	0,00007	0,0051	0,99303	34
1548/774/774/516	1,101	0,008	5,378	0,0019	0,00009	0,0056	0,99147	34
1548/774/774/774	1,290	0,010	10,986	0,0022	0,00009	0,0058	0,97682	27

Tabelle A.15 fortgesetzt.

Neuronen pro Schicht	MA-PE Δp (%)	MAE PRL (bar)	MAE \dot{m} (t/h)	MAE	MSE	RM-SE	R ²	Score
1548/1032/516/516	1,022	0,007	6,005	0,0019	0,00008	0,0056	0,98979	34
1548/1032/516/774	1,358	0,010	7,949	0,0018	0,00007	0,0051	0,98607	30
1548/1032/774/516	1,633	0,011	8,491	0,0020	0,00007	0,0052	0,98326	29
1548/1032/774/774	0,983	0,011	9,297	0,0019	0,00125	0,0214	0,92381	18
1634/1118/860/860	0,851	0,008	3,171	0,0016	0,00008	0,0058	0,98212	34
1634/1118/860/1118	1,299	0,008	6,651	0,0018	0,00007	0,0051	0,98939	33
1634/1118/1118/860	1,092	0,006	3,613	0,0014	0,00021	0,0091	0,98321	31
1634/1118/1118/1118	1,329	0,012	12,390	0,0020	0,00008	0,0060	0,98513	29
1634/1376/860/860	0,895	0,006	3,635	0,0017	0,00007	0,0051	0,97665	33
1634/1376/860/1118	0,891	0,008	11,758	0,0018	0,00007	0,0049	0,98844	31
1634/1376/1118/860	0,826	0,007	7,826	0,0016	0,00007	0,0048	0,99139	32
1634/1376/1118/1118	0,801	0,016	6,310	0,0018	0,00007	0,0048	0,99371	31
1892/1118/860/860	1,404	0,019	13,889	0,0023	0,00008	0,0060	0,98695	27
1892/1118/860/1118	0,896	0,005	4,239	0,0014	0,00007	0,0045	0,98261	34
1892/1118/1118/860	0,778	0,006	6,473	0,0014	0,00006	0,0047	0,98210	32
1892/1118/1118/1118	0,916	0,008	5,344	0,0020	0,00034	0,0145	0,96977	25
1892/1376/860/860	1,061	0,011	10,023	0,0018	0,00018	0,0093	0,98365	27
1892/1376/860/1118	1,067	0,005	6,039	0,0016	0,00007	0,0048	0,98107	33
1892/1376/1118/860	1,362	0,006	5,004	0,0017	0,00007	0,0048	0,98680	34
1892/1376/1118/1118	0,765	0,007	6,349	0,0017	0,00097	0,0150	0,91786	21
2150/1118/860/860	0,887	0,007	6,082	0,0016	0,00007	0,0050	0,97986	33
2150/1118/860/1118	1,143	0,012	7,880	0,0018	0,00007	0,0049	0,98301	30
2150/1118/1118/860	1,045	0,012	4,507	0,0018	0,00007	0,0050	0,98130	32
2150/1376/860/860	0,908	0,007	5,811	0,0015	0,00007	0,0048	0,98536	33
2150/1376/860/1118	0,706	0,006	4,713	0,0014	0,00006	0,0047	0,98559	33
2150/1376/1118/860	0,779	0,013	5,225	0,0016	0,00007	0,0047	0,97936	31

Tabelle A.16.: Bewertungsscore neuronaler Netze für Fernwärmenetz 3.

	MAPE Δp (%)	MAE PRL (bar)	MAE \dot{m} (t/h)	MAE	MSE	RMSE	R ²
Minimum	0,706	0,005	3,171	0,0014	0,00006	0,0045	0,91786
Maximum	7,957	0,042	18,266	0,0087	0,00125	0,0214	0,99371
Punkte							
5	< 1,431	< 0,008	< 4,680	< 0,0022	< 0,00018	< 0,0062	> 0,98613
4	< 2,156	< 0,012	< 6,190	< 0,0029	< 0,00030	< 0,0079	> 0,97854
3	< 2,881	< 0,016	< 7,699	< 0,0036	< 0,00042	< 0,0096	< 0,97096
2	< 3,606	< 0,020	< 9,209	< 0,0043	< 0,00054	< 0,0113	> 0,96337
1	Rest	Rest	Rest	Rest	Rest	Rest	Rest

Die folgende Tabelle A.17 beschreibt alle Ausgangsgrößen der trainierten neuronalen Netze. Zudem wird angegeben, ob die Ausgangsgröße in die Berechnung der relativen Modellgenauigkeit in Kapitel 6.3.2 einfließt. Einige Ausgänge werden in der späteren Optimierung (OPT) nicht mehr verwendet und sind daher nicht relevant. Andere Modellausgänge verursachen nur hohe relative Abweichungen. Zum

Beispiel die Heizwerke und die Druckerhöhungsanlage sind in den Simulationen häufig ausgestellt, weshalb die Sollwerte nach $SIR\ 3S^{\circ} 0$ entsprechen, das neuronale Netz aber kleine absolute Werte ungleich 0 vorhersagt. Somit führen hier absolut kleine Werte zu hohen relativen Abweichungen.

Tabelle A.17.: Ausgangsgrößen der neuronalen Netze für Fernwärmenetz 3 und Verwendung für die Berechnung der relativen Modellgenauigkeit.

Index	Modellausgang	Für Auswertung der Modellgenauigkeit verwendet?	Warum nicht?
0	P HKW	Nur für Beispieltage	In Testdaten Fälle mit Leistung = 0 vorhanden.
1	\dot{m} TG 1	Ja	
2	\dot{m} TG 2	Ja	
3	\dot{m} HKW	Nur für Beispieltage	In Testdaten Fälle mit Leistung = 0 vorhanden.
4	\dot{m} HW 1	Für Beispieltage im Winter / Sommer	Werte oft nahe 0.
5	\dot{m} HW 3	Nur für Beispieltage	In Testdaten Fälle mit Leistung = 0 vorhanden.
6	\dot{m} HW 2	Nein	Werte oft nahe 0.
7	\dot{m} DEA	Nein	Werte oft nahe 0.
8	n_{Pumpe} RL HKW	Ja	
9	n_{Pumpe} RL HW 1	Nein	Werte oft nahe 0.
10	n_{Pumpe} VL HW 3	Ja	
11	n_{Pumpe} RL HW 2	Nein	Werte oft nahe 0.
12	n_{Pumpe} VL DEA	Nein	Werte oft nahe 0.
13	n_{Pumpe} RL DEA	Nein	Werte oft nahe 0.
14	P_{Pumpe} RL HKW	Nein	Nicht für OPT nötig.
15	P_{Pumpe} RL HW 1	Nein	Nicht für OPT nötig.
16	P_{Pumpe} VL HW 3	Nein	Nicht für OPT nötig.
17	P_{Pumpe} RL HW 2	Nein	Nicht für OPT nötig.
18	P_{Pumpe} VL DEA	Nein	Nicht für OPT nötig.
19	P_{Pumpe} RL DEA	Nein	Nicht für OPT nötig.
20	v HKW	Nur für Beispieltage	In Testdaten Fälle mit Leistung = 0 vorhanden.
21	v HW 1	Für Beispieltage im Winter / Sommer	Werte oft nahe 0.
22	v HW 3	Nur für Beispieltage	In Testdaten Fälle mit Leistung = 0 vorhanden.
23	v HW 2	Nein	Werte oft nahe 0.
24	v DEA	Nein	Werte oft nahe 0.
25	Min. Δp TG 1	Ja	
26	Min. Δp TG 2	Ja	
27	Max. p_{RL} TG 1	Ja	
28	Max. p_{RL} TG 2	Ja	
29	Min. p_{RL} TG 1	Ja	
30	Min. p_{RL} TG 2	Ja	
31	Max. p_{VL} TG 1	Ja	
32	Max. p_{VL} TG 2	Ja	

Tabelle A.17 fortgesetzt.

Index	Modellausgang	Für Auswertung der Modellgenauigkeit verwendet?	Warum nicht?
33	Min. T_{VL} TG 1	Für Beispieltage im Winter / Frühling	Mittlerweile behobene, fehlerhafte Ausgabe des Interfaces zu <i>SIR 3S</i> [®] .
34	Min. T_{VL} TG 2	Nein	Mittlerweile behobene, fehlerhafte Ausgabe des Interfaces zu <i>SIR 3S</i> [®] .
35	Wärmeverbrauch TG 1	Nein	Nicht für OPT nötig.
36	Wärmeverbrauch TG 2	Nein	Nicht für OPT nötig.
37	Wärmeverlust	Nein	Nicht für OPT nötig.
38	Fehlmassenstrom	Nein	Werte oft nahe 0.
39	p_{VL} HW 3	Ja	
40	p_{RL} HW 3	Ja	
41	p_{VL} HKW	Ja	
42	p_{RL} HKW	Ja	

In den Tabellen A.18 bis A.20 werden die in Kapitel 6.3.2 gewählten neuronalen Netze für die drei Beispieltage ausgewertet. Trainiert wird jeweils ein Netz mit der Rücklauftemperatur als Eingang und ein Netz ohne die Temperatur. Das Netz mit der Rücklauftemperatur besteht aus vier versteckten Schichten mit 258, 258, 172 und 43 Neuronen. Für das Netz ohne die Rücklauftemperatur wird eine Struktur aus drei Schichten mit 215, 172 und 172 Neuronen gewählt. Dargestellt sind die maximalen absoluten und relativen Abweichungen sowie der MAPE. Alle relativen Abweichungen größer 3% sind rot markiert. Einige der Abweichungen sind grau markiert, falls die hohen relativen Abweichungen lediglich durch Ausgangswerte nahe 0 entstehen oder wie bei der minimalen Vorlauftemperatur durch inzwischen behobene Berechnungsfehler in *SIR 3S*[®].

Tabelle A.18.: Maximale absolute und relative Abweichungen der neuronalen Netze zu *SIR 3S*[®] für Netz 3 und für zehn Simulationstage basierend auf Beispieltag 1 im Februar.

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	MA-PE mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	MA-PE ohne T_{RL} (%)
P HKW (MW)	0,628	0,456	0,072	0,761	0,468	0,105
\dot{m} TG 1 (t/h)	39,173	0,708	0,119	373,067	8,162	1,0233
\dot{m} TG 2 (t/h)	16,570	3,077	0,196	28,611	7,037	1,072
\dot{m} HKW (t/h)	19,903	0,750	0,078	164,533	6,893	0,972
\dot{m} HW 1 (t/h)	7,862	0,303	0,162	189,051	8,457	1,041
\dot{m} HW 3 (t/h)	2,149	0,608	0,227	23,231	6,653	1,084
\dot{m} HW 2 (t/h)	0,586	inf	inf	1,234	inf	inf
\dot{m} DEA (t/h)	17,327	445,667	3,828	11,821	1716,496	7,167
n_{Pumpe} RL HKW (min^{-1})	17,075	1,797	0,109	40,754	4,439	0,601
n_{Pumpe} RL HW 1 (min^{-1})	15,056	1,504	0,101	49,436	5,502	0,710
n_{Pumpe} VL HW 3 (min^{-1})	5,008	0,742	0,197	11,755	1,857	0,494
n_{Pumpe} RL HW 2 (min^{-1})	2,419	inf	inf	6,746	inf	inf
n_{Pumpe} VL DEA (min^{-1})	94,703	inf	inf	111,802	inf	inf

Tabelle A.18 fortgesetzt.

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	MA-PE mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	MA-PE ohne T_{RL} (%)
n_{Pumpe} RL DEA (min^{-1})	38,930	inf	inf	34,001	inf	inf
P_{Pumpe} RL HKW (kW)	23,246	5,021	0,251	55,505	13,096	2,011
P_{Pumpe} RL HW 1 (kW)	19,623	4,014	0,260	61,245	17,682	2,190
P_{Pumpe} VL HW 3 (kW)	1,128	1,793	0,465	5,260	10,566	2,396
P_{Pumpe} RL HW 2 (kW)	0,257	inf	inf	0,448	inf	inf
P_{Pumpe} VL DEA (kW)	0,591	inf	inf	1,430	inf	inf
P_{Pumpe} RL DEA (kW)	1,175	inf	inf	4,853	inf	inf
v HKW (m/s)	0,007	0,507	0,122	0,132	7,183	0,987
v HW 1 (m/s)	0,007	0,389	0,169	0,149	8,722	1,043
v HW 3 (m/s)	0,003	0,621	0,308	0,034	6,560	1,118
v HW 2 (m/s)	0,002	inf	inf	0,004	inf	inf
v DEA (m/s)	0,040	304,296	3,516	0,030	958,145	5,694
Min. Δp TG 1 (bar)	0,236	10,098	0,398	0,224	9,568	0,645
Min. Δp TG 2 (bar)	0,059	3,272	0,347	0,054	2,973	0,331
Max. p_{RL} TG 1 (bar)	0,032	0,505	0,093	0,130	2,089	0,431
Max. p_{RL} TG 2 (bar)	0,168	2,597	0,189	0,226	3,503	0,459
Min. p_{RL} TG 1 (bar)	0,011	0,298	0,098	0,012	0,311	0,127
Min. p_{RL} TG 2 (bar)	0,164	4,542	0,311	0,244	6,756	0,916
Max. p_{VL} TG 1 (bar)	0,251	2,651	0,142	0,359	3,967	0,541
Max. p_{VL} TG 2 (bar)	0,218	2,651	0,166	0,237	2,969	0,404
Min. T_{VL} TG 1 ($^{\circ}\text{C}$)	6,999	6,926	0,300	7,052	6,979	0,376
Min. T_{VL} TG 2 ($^{\circ}\text{C}$)	43,842	80,277	0,492	44,044	80,647	0,533
Wärmeverbrauch TG 1 (MW)	0,035	0,182	0,030	0,552	2,778	0,307
Wärmeverbrauch TG 2 (MW)	1,822	0,520	0,111	1,722	0,491	0,088
Wärmeverlust (MW)	0,231	0,812	0,250	0,261	0,891	0,135
Fehlmassenstrom (t/h)	29,268	inf	126,160	29,792	inf	354,601
p_{VL} HW 3 (bar)	0,215	2,872	0,183	0,228	2,981	0,508
p_{RL} HW 3 (bar)	0,165	4,547	0,281	0,243	6,729	0,906
p_{VL} HKW (bar)	0,253	2,851	0,156	0,336	3,883	0,607
p_{RL} HKW (bar)	0,005	0,134	0,072	0,010	0,205	0,117

Die Auswertung für das neuronale Netz mit Rücklauftemperatureingang soll hier noch detaillierter beschrieben werden. Die Abweichungen für das Netz ohne Rücklauftemperatur sind ähnlich zu beurteilen, nur allgemein höher. Das Netz mit der Temperatur zeigt an Beispieltag 1 in Tabelle A.18 zunächst für den Massenstrom in Teilgebiet 2 eine maximale relative Abweichung über 3%. Im Durchschnitt ist diese mit 0,196% sehr gering. Der Massenstrom, die Pumpendrehzahl und -leistung sowie die Strömungsgeschwindigkeit von HW 2 weichen relativ gesehen stark ab. Das Heizwerk wird an dem Beispieltag allerdings nicht verwendet. Die Abweichungen sind absolut sehr gering. Der Massenstrom und die Strömungsgeschwindigkeit der DEA wird absolut gesehen gut vorhergesagt, nahe dem Nullpunkt kommt es aber wieder zu hohen relativen Fehlern. Die Pumpendrehzahlen der DEA können nicht sehr gut berechnet werden. Allerdings befindet sich die Anlage hier außerhalb ihres Betriebsbereichs und

unterhalb der minimalen Drehzahl. Diese wird an allen Beispieltagen korrekt bestimmt, sodass dem Modell in einer Optimierung eine fixe Drehzahl vorgegeben werden kann. Hohe relative Fehler treten ebenfalls für alle Pumpenleistungen auf. Im Schnitt sind die Abweichungen aber absolut und relativ gering. Die Pumpenleistungen werden zudem für die in dieser Arbeit durchgeführten Optimierungen nicht benötigt. Einige Druckausgänge weisen zeitweise relative Fehler über 3% auf. Auch hier sind die absoluten und durchschnittlichen relativen Fehler gering. Für die minimalen Vorlauftemperaturen entstehen Abweichungen durch einen inzwischen behobenen Fehler in *SIR 3S*[®] auf. Der Fehlmassenstrom kann nicht gut vorhergesagt werden. Wie in Kapitel 6.3.2 beschrieben, ist die Einordnung in einen zu hohen oder akzeptablen Fehlmassenstrom allerdings in allen Simulationen und Optimierungen erfolgreich. Insgesamt kann das neuronale Netz Wintertage somit ausreichend genau für eine Optimierung berechnen. Die Ergebnisse sind größtenteils sehr gut. Einige der hier ausgewerteten Ausgänge können aber durch weitere Trainingsdaten verbessert werden.

Tabelle A.19.: Maximale absolute und relative Abweichungen der neuronalen Netze zu *SIR 3S*[®] für Netz 3 und für zehn Simulationstage basierend auf Beispieltag 2 im Mai.

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	<i>MAPE</i> mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	<i>MAPE</i> ohne T_{RL} (%)
P HKW (MW)	0,498	0,681	0,215	1,009	1,333	0,366
\dot{m} TG 1 (t/h)	25,121	1,834	0,671	153,933	8,442	2,851
\dot{m} TG 2 (t/h)	18,140	9,566	1,256	57,566	25,985	4,146
\dot{m} HKW (t/h)	8,359	0,638	0,114	118,151	5,983	1,518
\dot{m} HW 1 (t/h)	21,383	inf	inf	49,559	inf	inf
\dot{m} HW 3 (t/h)	5,647	3,503	1,403	39,132	31,746	5,924
\dot{m} HW 2 (t/h)	0,680	inf	inf	2,209	inf	inf
\dot{m} DEA (t/h)	15,081	1753,668	4,153	19,989	1369,710	5,694
n_{Pumpe} RL HKW (min^{-1})	7,913	1,059	0,136	28,631	3,427	0,740
n_{Pumpe} RL HW 1 (min^{-1})	617,755	inf	inf	88,740	inf	inf
n_{Pumpe} VL HW 3 (min^{-1})	8,238	1,756	0,162	9,740	2,144	0,500
n_{Pumpe} RL HW 2 (min^{-1})	2,585	inf	inf	7,289	inf	inf
n_{Pumpe} VL DEA (min^{-1})	19,213	inf	inf	43,886	inf	inf
n_{Pumpe} RL DEA (min^{-1})	3,669	inf	inf	3,751	inf	inf
P_{Pumpe} RL HKW (kW)	8,567	5,247	2,490	36,030	11,921	2,919
P_{Pumpe} RL HW 1 (kW)	10,751	inf	inf	20,443	inf	inf
P_{Pumpe} VL HW 3 (kW)	1,355	11,206	2,320	2,646	20,660	5,510
P_{Pumpe} RL HW 2 (kW)	0,462	inf	inf	0,438	inf	inf
P_{Pumpe} VL DEA (kW)	0,285	inf	inf	1,210	inf	inf
P_{Pumpe} RL DEA (kW)	2,044	inf	inf	3,067	inf	inf
v HKW (m/s)	0,006	0,388	0,128	0,086	6,384	1,553
v HW 1 (m/s)	0,015	inf	inf	0,038	inf	inf
v HW 3 (m/s)	0,008	3,169	1,233	0,056	30,849	5,595
v HW 2 (m/s)	0,002	inf	inf	0,004	inf	inf
v DEA (m/s)	0,035	1835,018	3,866	0,048	1126,067	7,377
Min. Δp TG 1 (bar)	0,072	3,662	0,635	0,059	2,848	0,987
Min. Δp TG 2 (bar)	0,025	1,374	0,477	0,042	2,339	0,549
Max. p_{RL} TG 1 (bar)	0,032	0,518	0,085	0,119	2,053	0,357
Max. p_{RL} TG 2 (bar)	0,025	0,424	0,197	0,141	2,333	0,400
Min. p_{RL} TG 1 (bar)	0,038	0,921	0,194	0,033	0,821	0,208
Min. p_{RL} TG 2 (bar)	0,031	0,862	0,136	0,143	3,385	0,674

Tabelle A.19 fortgesetzt.

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	$MAPE$ mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	$MAPE$ ohne T_{RL} (%)
Max. p_{VL} TG 1 (bar)	0,044	0,577	0,107	0,206	2,469	0,441
Max. p_{VL} TG 2 (bar)	0,050	0,696	0,084	0,137	1,880	0,485
Min. T_{VL} TG 1 (°C)	16,594	19,821	6,986	23,425	27,979	8,099
Min. T_{VL} TG 2 (°C)	59,865	595,927	9,273	59,572	588,990	10,420
Wärmeverbrauch TG 1 (MW)	0,075	0,442	0,174	0,366	2,184	0,493
Wärmeverbrauch TG 2 (MW)	1,548	3,080	1,471	2,397	4,242	2,083
Wärmeverlust (MW)	0,161	4,812	0,918	0,319	8,607	2,796
Fehlmassenstrom (t/h)	11,522	375,815	151,936	15,239	1261,251	204,380
p_{VL} HW 3 (bar)	0,055	1,012	0,148	0,172	2,832	0,463
p_{RL} HW 3 (bar)	0,032	0,822	0,136	0,133	3,111	0,590
p_{VL} HKW (bar)	0,051	0,705	0,078	0,251	3,071	0,549
p_{RL} HKW (bar)	0,012	0,338	0,072	0,012	0,247	0,082

Tabelle A.19 zeigt die Berechnungsfehler für Beispieltag 2 im Mai. Verglichen mit Tag 1 entstehen höhere relative Abweichungen. Vor allem für HW 1 sind häufig sehr hohe relative Abweichungen bei geringen absoluten Fehlern zu beobachten. Der Grund ist, dass HW 1 den Großteil von Beispieltag 2 nicht verwendet wird. Für die Drehzahl der Rücklaufpumpen sind auch absolut sehr hohe Abweichungen sichtbar. Hier ergeben sich in $SIR\ 3S^{\circledast}$ hohe Drehzahlen der Pumpen trotz Aus-Zustand des HW. In der Berechnung der neuronalen Netze wird eine Drehzahl nahe 0 vorausgesagt. Im Aus-Zustand des HW ist die Drehzahl der Pumpen für eine spätere Optimierung irrelevant, weshalb Berechnungsfehler nicht kritisch sind. Im An-Zustand werden die Drehzahlen korrekt berechnet. Für die restlichen Ausgangsgrößen ergeben sich keine wesentlichen Veränderungen, verglichen mit der Fehlerauswertung für Beispieltag 1.

Auch für Beispieltag 3 in Tabelle A.20 können die Ergebnisse ähnlich interpretiert werden. Die relativen Abweichungen nehmen aufgrund absolut sehr niedriger Werte etwas weiter zu. Kritisch ist hier allerdings die Simulation der Differenzdrücke und maximalen Vorlaufdrücke in Teilgebiet 2. Diese weichen auch absolut zu einigen Zeitpunkten um bis zu 0,4 bar ab. Diese fehlerhafte Berechnung ist für eine Optimierung problematisch. Hier sollten die neuronalen Netze mit weiteren Trainingsdaten von Tagen mit niedrigen Wärmeverbräuchen verbessert werden.

Tabelle A.20.: Maximale absolute und relative Abweichungen der neuronalen Netze zu $SIR\ 3S^{\circledast}$ für Netz 3 und für zehn Simulationstage basierend auf Beispieltag 3 im Juli.

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	$MAPE$ mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	$MAPE$ ohne T_{RL} (%)
P HKW (MW)	0,452	3,085	0,669	1,186	8,189	1,312
\dot{m} TG 1 (t/h)	18,827	1,039	0,512	174,204	9,246	2,541
\dot{m} TG 2 (t/h)	10,525	3,567	1,203	74,226	29,170	6,540
\dot{m} HKW (t/h)	7,391	1,104	0,264	103,689	9,089	2,280
\dot{m} HW 1 (t/h)	5,841	5,330	0,557	97,789	55,865	4,549
\dot{m} HW 3 (t/h)	5,990	1,719	0,486	58,153	17,132	4,840
\dot{m} HW 2 (t/h)	0,457	inf	inf	2,555	inf	inf

Tabelle A.20 fortgesetzt.

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	$MAPE$ mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	$MAPE$ ohne T_{RL} (%)
\dot{m} DEA (t/h)	12,941	39,402	5,372	27,366	56,590	13,319
n_{Pumpe} RL HKW (min^{-1})	5,356	0,948	0,186	21,964	3,479	0,741
n_{Pumpe} RL HW 1 (min^{-1})	5,165	0,949	0,228	62,668	11,541	0,929
n_{Pumpe} VL HW 3 (min^{-1})	32,097	5,538	1,510	43,227	7,258	2,334
n_{Pumpe} RL HW 2 (min^{-1})	1,182	inf	inf	5,275	inf	inf
n_{Pumpe} VL DEA (min^{-1})	1,991	inf	inf	2,606	inf	inf
n_{Pumpe} RL DEA (min^{-1})	38,553	inf	inf	99,919	inf	inf
P_{Pumpe} RL HKW (kW)	8,660	13,459	4,955	15,994	19,674	5,183
P_{Pumpe} RL HW 1 (kW)	6,271	46,333	7,463	20,331	108,197	17,942
P_{Pumpe} VL HW 3 (kW)	7,305	17,982	4,235	10,225	23,271	8,369
P_{Pumpe} RL HW 2 (kW)	0,298	inf	inf	0,448	inf	inf
P_{Pumpe} VL DEA (kW)	0,243	inf	inf	0,269	inf	inf
P_{Pumpe} RL DEA (kW)	2,702	inf	inf	3,665	inf	inf
v HKW (m/s)	0,010	2,457	0,702	0,088	10,281	2,341
v HW 1 (m/s)	0,009	11,769	2,033	0,073	52,625	4,410
v HW 3 (m/s)	0,009	1,793	0,491	0,084	16,952	4,909
v HW 2 (m/s)	0,003	inf	inf	0,002	inf	inf
v DEA (m/s)	0,027	36,122	4,754	0,064	58,838	13,599
Min. Δp TG 1 (bar)	0,025	1,409	0,536	0,062	3,571	1,135
Min. Δp TG 2 (bar)	0,406	28,219	5,769	0,274	15,190	3,472
Max. p_{RL} TG 1 (bar)	0,023	0,369	0,097	0,054	0,887	0,264
Max. p_{RL} TG 2 (bar)	0,052	0,858	0,214	0,131	2,133	0,675
Min. p_{RL} TG 1 (bar)	0,047	0,978	0,294	0,054	1,145	0,309
Min. p_{RL} TG 2 (bar)	0,051	1,409	0,326	0,189	5,720	1,243
Max. p_{VL} TG 1 (bar)	0,036	0,449	0,111	0,119	1,482	0,372
Max. p_{VL} TG 2 (bar)	0,371	4,742	1,068	0,306	3,744	0,610
Min. T_{VL} TG 1 ($^{\circ}\text{C}$)	58,713	484,309	5,683	57,038	470,495	5,604
Min. T_{VL} TG 2 ($^{\circ}\text{C}$)	24,565	57,316	3,454	22,033	51,407	3,840
Wärmeverbrauch TG 1 (MW)	0,061	0,372	0,138	0,450	2,789	0,686
Wärmeverbrauch TG 2 (MW)	2,227	5,490	2,843	1,780	3,774	1,431
Wärmeverlust (MW)	0,177	5,175	0,914	0,364	6,858	3,453
Fehlmassenstrom (t/h)	1,418	1164,41	177,869	16,348	13429,3	530,278
p_{VL} HW 3 (bar)	0,424	6,293	1,340	0,378	5,360	1,283
p_{RL} HW 3 (bar)	0,050	1,373	0,332	0,193	5,778	1,261
p_{VL} HKW (bar)	0,030	0,419	0,107	0,179	2,461	0,521
p_{RL} HKW (bar)	0,008	0,154	0,082	0,006	0,118	0,052

A.1.5. Modellbildung und -auswertung - Fernwärmenetz 4

In diesem Teil des Anhangs wird die Modellbildung und -auswertung in *SIR 3S*[®] und mit neuronalen Netzen für das in Kapitel 6.4 vorgestellte Fernwärmenetz 4 durchgeführt. Das Vorgehen erfolgt dabei analog zu dem für Fernwärmenetz 3 in Kapitel 6.3.

Netz 4: *SIR 3S*[®]

Fernwärmenetz 4 wird in *SIR 3S*[®] im vollständigen Zustand mit 16623 Knoten, 15703 Rohrleitungen, vereinfachten Stationsmodellierungen und quasi-stationären Simulationen verwendet. Das Modell ist mit Beispieltag 1 (Hochlastfall im Februar 2021) und Beispieltag 2 (Mai 2020) validiert worden. Der Niedriglastfall im Juli 2021 wird aufgrund fehlerhafter Prozessdaten nicht verwendet. Die Eingangsgrößen des Modells sind im Folgenden beschrieben:

- Massenströme der drei Teilgebiete
- Massenstrom und Vorlauftemperatur des HW
- Vorlauftemperatur des HKW. Der Massenstrom ergibt sich aus der in *SIR 3S*[®] implementierten Differenzdruckregelung.
- An / Aus Zustand der Vor- und Rücklaufpumpen der DEAs
- Solldruck der Druckhaltung in der Speicheranlage
- Beimischung für Teilgebiet 3. Hier wird ein kalter Massenstrom aus dem Rücklauf beigemischt, um die Vorlauftemperatur zu senken.
- Speicher Belade- oder Entlademassenstrom
- Speicher Vor- und Rücklauftemperatur
- Rücklauftemperaturen der drei Teilgebiete

Das Fernwärmenetzmodell enthält bereits eine Regelung der Differenzdrücke. Andere Regelungen für beispielsweise die DEAs, die Druckhaltung der Speicheranlage und der Beimischung sind nicht in dem Modell implementiert. Diese können entweder durch eine Optimierung übernommen oder vor einem Einsatz in das Modell nachgearbeitet werden. Nach der Validierung ergeben sich die mittleren 24 Stunden Druck- und Temperaturabweichungen zu den verfügbaren Messwerten in Tabelle A.21.

Tabelle A.21.: Mittlere 24h-Abweichungen der Nachberechnung von Fernwärmenetz 4 in *SIR 3S*[®] an den drei Beispieltagen.

	Tag 1	Tag 2
Druck (bar)	0,23	0,18
Standardabweichung Druck (bar)	0,20	0,14
Druck (%)	4,0	2,9
Temperatur (°C)	2,96	3,07
Standardabweichung Temperatur (°C)	3,79	2,96
Temperatur (%)	3,2	4,0

Netz 4: Neuronale Netze

Basierend auf den Erkenntnissen für Fernwärmenetz 1 bis 3 werden für Netz 4 neuronale Netze trainiert, um das Netzverhalten abzubilden und Rechenzeiten zu reduzieren.

Datenerzeugung

Für das Modelltraining werden 745 930 stationäre Simulationen mit variierenden Eingangsgrößen durchgeführt. Nach dem Aussortieren von Datensätzen mit hohen Fehlmassenströmen und Pumpenleistungen kleiner 0 min^{-1} bleiben 685 340 stationäre Simulationen übrig. Die Eingangsgrößen werden wie für Netz 2 und 3 zwischen den Minimal- und Maximalwerten der Beispieltage variiert. Hier sind allerdings nur zwei Beispieltage vorhanden und ein Minimallastfall im Sommer fehlt. Massenströme und Vorlauftemperaturen der Wärmeerzeugungsanlagen werden daher in der Datenerzeugung deutlich weiter abgesenkt als an den Beispieltagen. Zusätzlich werden höhere und niedrigere Be- und Entladungsmassenströme der Speicheranlage simuliert. Allerdings werden Druckhaltung, Beimischungsmassenstrom, Speicher Vor- und Rücklauftemperaturen sowie die Rücklauftemperaturen der Wärmeerzeugungsanlagen nur entsprechend den zwei Beispieltagen simuliert. Die Anwendbarkeit auf einen Niedriglastfall im Sommer muss noch weiter untersucht werden. Zudem werden an den Beispieltagen nur die Rücklaufpumpen der DEAs verwendet. Andere Einstellungen der DEAs sind nicht in den Simulationsdaten enthalten.

Für das Training von neuronalen Netzen werden die Gesamtleistungen der drei Teilgebiete und Leistungen des HW sowie der Speicher statt deren Massenströmen verwendet. So können die optimierten Sollwerte der wirtschaftlichen Optimierung direkt in ein Modell gegeben werden. Die Massenströme sind anschließend ein Ausgang der Modelle. Die folgenden 15 Eingänge werden für die zu trainierenden Modelle verwendet:

- Gesamte Wärmeleistung und relativer Wärmeverbrauch der Teilgebiete 1 und 2
- Leistung des HW sowie Vorlauftemperaturen des HW und HKW
- Sollwert für die Druckhaltung
- Kalte Beimischung für die Vorlauftemperatur in Teilgebiet 3
- Speicher Be- und Entladeleistung sowie Vor- und Rücklauftemperaturen
- Rücklauftemperaturen der drei Teilgebiete. Hier werden wieder Modelle mit und ohne den Temperaturen trainiert.

Für eine Optimierung mit 24-Stunden-Prognose ergeben sich hier, verglichen mit den bisherigen Fernwärmenetzen weitere Probleme. Nicht nur die Rücklauftemperaturen der Teilgebiete sind für die Zukunft nicht bekannt, sondern auch die Vor- und Rücklauftemperaturen der Speicher müssen vorhergesagt oder angenommen werden. Die kalte Beimischung für die Vorlauftemperatur in Teilgebiet 3 muss so eingestellt werden, dass ein Temperatursollwert erreicht wird.

Die in der folgenden Tabelle A.22 gelisteten 58 Werte werden in der Optimierung als Zielgrößen oder Restriktionen verwendet und sind somit Ausgänge der neuronalen Netze. Nur 43 Ausgänge werden anschließend für die Bewertung der relativen Modellgenauigkeit verwendet und sind in der Tabelle gekennzeichnet. Größen, die später in der Optimierung nicht mehr notwendig sind, können vernachlässigt werden. Zudem sind einige Werte, wie zum Beispiel der Massenstrom der Speicherbeladung und -entladung häufig 0. Die neuronalen Netze sagen hier nicht immer genau 0 voraus, weshalb selbst sehr kleine absolute Abweichungen zu hohen relativen Fehlern führen. Diese sollen nicht berücksichtigt werden. Zudem funktioniert die Vorhersage des Speicherinhalts mit dem bisher gewählten stationären Ansatz nicht. Hier wäre für eine korrekte Vorhersage jeweils der Füllstand des vorherigen Zeitschritts

notwendig. Diese Art der Vorhersage wurde aber noch nicht implementiert. Der Ausgang ist somit nicht verwendbar.

Tabelle A.22.: Ausgangsgrößen der neuronalen Netze für Fernwärmenetz 3 und Verwendung für die Berechnung der relativen Modellgenauigkeit.

Index	Modellausgang	Für Auswertung der Modellgenauigkeit verwendet?	Warum nicht?
0	Speicherinhalt	Nein	Vorhersage funktioniert mit stationärem Ansatz nicht.
1	P HKW	Ja	
2	\dot{m} Speicheranlage TG 3	Ja	
3	\dot{m} Speicheranlage TG 1	Nein	Werte oft nahe 0.
4	\dot{m} Speicheranlage TG 2	Ja	
5	\dot{m} Ventil Speicheranlage	Ja	
6	\dot{m} HW	Nein	Werte oft nahe 0.
7	\dot{m} HKW	Ja	
8	\dot{m} DEA 2	Ja	
9	\dot{m} DEA 1	Ja	
10	\dot{m} Speicherbeladung	Nein	Werte oft nahe 0.
11	\dot{m} Speicherentladung	Nein	Werte oft nahe 0.
12	n_{Pumpe} RL DEA 2	Ja	
13	n_{Pumpe} RL DEA 1	Ja	
14	n_{Pumpe} RL TG 3	Ja	
15	n_{Pumpe} RL TG 1	Ja	
16	n_{Pumpe} VL TG 2	Nein	Werte oft nahe 0.
17	n_{Pumpe} RL HKW	Ja	
18	n_{Pumpe} RL HW	Nein	Werte oft nahe 0.
19	P_{Pumpe} RL DEA 2	Nein	Nicht für OPT nötig.
20	P_{Pumpe} RL DEA 1	Nein	Nicht für OPT nötig.
21	P_{Pumpe} RL TG 3	Nein	Nicht für OPT nötig.
22	P_{Pumpe} RL TG 1	Nein	Nicht für OPT nötig.
23	P_{Pumpe} VL TG 2	Nein	Nicht für OPT nötig.
24	P_{Pumpe} RL HKW	Nein	Nicht für OPT nötig.
25	P_{Pumpe} RL HW	Nein	Nicht für OPT nötig.
26	Fehlmassenstrom	Nein	Werte oft nahe 0.
27	Max. v	Ja	
28	Min. Δp TG 3	Ja	
29	Min. Δp TG 1	Ja	
30	Min. Δp TG 2	Ja	
31	Max. p_{RL} Schlechtpunkt (SP) TG 1	Ja	
32	Max. p_{RL} SP TG 2	Ja	
33	Min. p_{RL} TG 3	Ja	
34	Min. p_{RL} TG 1	Ja	
35	Min. p_{RL} TG 2	Ja	
36	Max. p_{VL} TG 3	Ja	
37	Max. p_{VL} TG 1	Ja	
38	Max. p_{VL} TG 2	Ja	

Tabelle A.22 fortgesetzt.

Index	Modellausgang	Für Auswertung der Modellgenauigkeit verwendet?	Warum nicht?
39	Min. p_{VL} SP DEA 2	Ja	
40	Min. p_{VL} SP DEA 1	Ja	
41	Min. p_{VL} SP TG 3	Ja	
42	Min. p_{VL} SP TG 1	Ja	
43	Min. p_{VL} SP TG 2	Ja	
44	Min. T_{VL} TG 3	Ja	
45	Min. T_{VL} TG 1	Ja	
46	Min. T_{VL} TG 2	Ja	
47	Wärmeverbrauch TG 3	Ja	
48	Wärmeverbrauch TG 2	Ja	
49	Wärmeverbrauch TG 1	Ja	
50	Wärmeverbrauch gesamt	Ja	
51	Wärmeverlust gesamt	Ja	
52	Wärmeverlust TG 3	Ja	
53	Wärmeverlust TG 2	Ja	
54	Wärmeverlust TG 1	Ja	
55	\dot{m} Speicheranlage TG 1	Ja	
56	\dot{m} Speicheranlage TG 2	Ja	
57	\dot{m} Speicheranlage TG 3	Ja	

Untersuchung der Anzahl der Schichten und Neuronen sowie der Lernrate

Mit den Daten werden neuronale Netze mit unterschiedlicher Anzahl an Schichten und Neuronen getestet, um eine möglichst hohe Genauigkeit verglichen mit *SIR 3S*[®] Simulationen zu erreichen. Hier werden basierend auf den Ergebnissen der vorherigen Netze nur vier Kombinationen aus Schichten und Neuronen abhängig von der Anzahl der Ausgänge für das Training verwendet. Die restlichen Einstellungen der neuronalen Netze verbleiben wie zuvor und sollen nicht erneut untersucht werden. Es wird also der Optimierer *Adam* mit einer Lernrate von 10^{-3} sowie einer alle 200 Epochen halbierten Lernrate (mit *Decay*) verwendet, die *Loss*-Funktion MAE, die Aktivierungsfunktion *Swish*, die Gewichte werden mit *He-Uniform* initialisiert und die Daten zufällig in 60 % Trainings-, 20 % Validierungs- und 20 % Testdaten unterteilt. Somit sind 411 204 Ein- und Ausgangsdatensätze für das Training und jeweils 137 068 Datensätze für die Validierung und das Testen vorhanden. Alle Daten werden zwischen 0 und 1 normiert. Für die Bewertung der neuronalen Netze werden die Kriterien MAPE und R^2 für 43 der 58 Ausgangsgrößen entsprechend Tabelle A.22 verwendet. Zusätzlich werden der durchschnittliche MAPE und MAE für die Druckausgänge der Netze berechnet. Eine vollständige Auswertung aller Ausgangsgrößen folgt später mit dem final gewählten neuronalen Netz. Zusätzlich wird die mit t_s bezeichnete durchschnittliche Rechenzeit der Modelle aus 100 in Schleife durchgeführten 24-Stunden-Simulationen im 15-Minuten-Takt bestimmt. Ein Rechner mit *Intel Core i7-4771* (4 x 3,50GHz) und 32 GB RAM wird verwendet. Die Ergebnisse sind in Tabelle A.23 dargestellt.

Tabelle A.23.: Bewertung neuronaler Netze mit Rücklaufftemperatureingang mit Testdaten für Fernwärmenetz 4 nach 1000 Epochen Training.

Neuronen pro Schicht	<i>MAPE</i> Drücke (%)	<i>MAE</i> Drücke (bar)	<i>MAPE</i> gesamt (%)	R^2	t_s (ms)
580/464/232	0,098	0,005	0,286	0,95153	0,805
232/116/58/29	0,174	0,010	0,623	0,93459	0,202
348/348/232/58	0,085	0,005	0,408	0,95434	0,543
464/580/464/348	0,084	0,005	0,323	0,95506	1,345
580/464/232 (mit <i>Decay</i>)	0,067	0,004	0,167	0,95292	0,794
232/116/58/29 (mit <i>Decay</i>)	0,176	0,010	0,612	0,93057	0,223
348/348/232/58 (mit <i>Decay</i>)	0,057	0,003	0,154	0,95511	0,547
464/580/464/348 (mit <i>Decay</i>)	0,053	0,003	0,137	0,95630	1,303

Modelle mit einem MAPE größer 0,5% und neuronale Netze mit einer Berechnungszeit über 1 ms werden bereits an dieser Stelle aussortiert. So bleiben für die weitere Auswertung vier neuronale Netze übrig, welche in Tabelle A.24 auf den Winter-Beispieltag 1 und in Tabelle A.25 auf den Maitag 2 angewandt werden. Für eine von den Trainings- und Testdaten unabhängige Bewertung der neuronalen Netze werden pro Beispieltag zehn 24-Stunden-Simulationen mit 15-Minuten-Zeitschritten und Variation der Eingangsdaten in *SIR 3S*[®] durchgeführt. Massenströme werden zufällig um bis zu $\pm 5\%$, Vorlauftemperaturen um bis zu $\pm 3^\circ\text{C}$ und Rücklaufftemperaturen um bis zu $\pm 1^\circ\text{C}$ verändert. Der Massenstrom der Beimischung in der Fernwärmanlage wird nur um bis zu $\pm 2\%$ variiert und die Druckhaltung wird konstant belassen. Diese zeigt an den Beispieltagen kaum Veränderungen. Auch die Einstellung der Pumpen der Druckerhaltungsanlagen wird konstant simuliert. Wie bereits erwähnt, können die neuronalen Netze so nur den An-Zustand der Rücklaufpumpen und Aus-Zustand der Vorlaufpumpen simulieren. Die zehn Tage werden anschließend mit neuronalen Netzen simuliert und die Ergebnisse im Folgenden verglichen. Die Bewertungskriterien für den Vergleich mit *SIR 3S*[®] sind die gleichen wie zuvor. Zusätzlich wird die maximale relative Abweichung für 43 der 58 Ausgangsgrößen angegeben.

Tabelle A.24.: Abweichung verschiedener neuronaler Netze für Fernwärmenetz 4 zu *SIR 3S*[®] für zehn Simulationstage basierend auf Beispieltag 1 im Februar.

Neuronen pro Schicht	<i>MAPE</i> Drücke (%)	<i>MAE</i> Drücke (bar)	<i>MAPE</i> gesamt (%)	Max. rel. Error (%)
580/464/232	0,120	0,008	0,170	31,497
348/348/232/58	0,084	0,005	0,145	8,560
580/464/232 (mit <i>Decay</i>)	0,076	0,005	0,080	4,446
348/348/232/58 (mit <i>Decay</i>)	0,082	0,005	0,085	13,205

Tabelle A.25.: Abweichung verschiedener neuronaler Netze für Fernwärmenetz 4 zu *SIR 3S*[®] für zehn Simulationstage basierend auf Beispieltag 2 im Mai.

Neuronen pro Schicht	<i>MAPE</i> Drücke (%)	<i>MAE</i> Drücke (bar)	<i>MAPE</i> gesamt (%)	Max. rel. Error (%)
580/464/232	0,290	0,017	1,300	32,442
348/348/232/58	0,132	0,009	0,999	37,441
580/464/232 (mit <i>Decay</i>)	0,135	0,009	0,523	13,436
348/348/232/58 (mit <i>Decay</i>)	0,210	0,012	0,947	39,332

Das mit abnehmender Lernrate trainierte neuronale Netz mit drei versteckten Schichten sowie 580, 464 und 232 Neuronen zeigt für beide Beispieltage die besten Ergebnisse und wird für die weitere Auswertung gewählt.

Neuronale Netze für Fernwärmenetz 4

Das Netz wird zusätzlich ohne die Rücklaufftemperaturen als Eingang trainiert. Die Lernkurven der beiden neuronalen Netze sind in Abbildung A.1 dargestellt. Zusammenfassend zeigt die folgende Tabelle A.26 die durchschnittlichen Abweichungen für Drücke und die entsprechend Tabelle A.22 relevanten Ausgangsgrößen für die Testdaten und zwei Beispieltage.

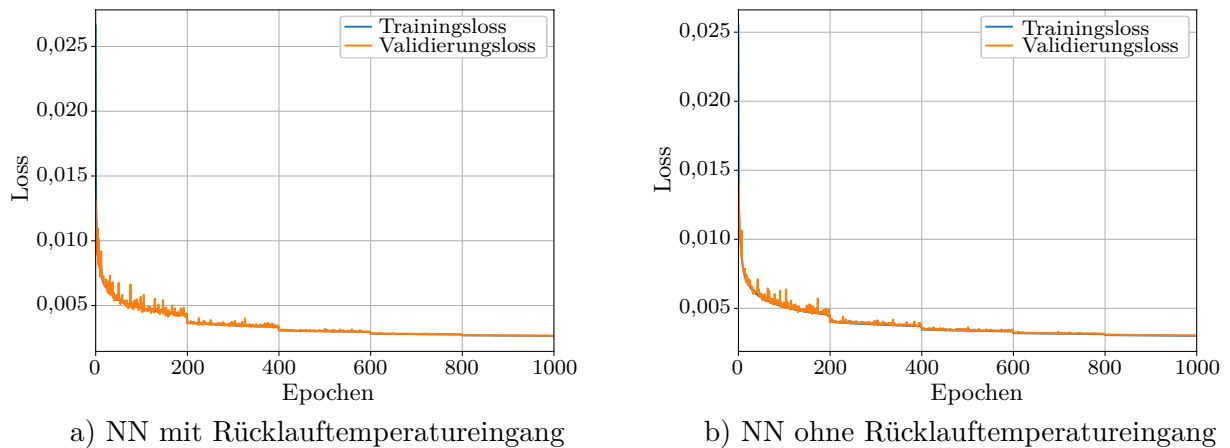


Abbildung A.1.: Lernkurven für die neuronalen Netze (NN) von Fernwärmenetz 4.

Tabelle A.26.: Durchschnittliche Abweichung der neuronalen Netze zu *SIR 3S*[®] für Fernwärmenetz 4

	MAPE Drücke (%)	MAE Drücke (bar)	MAPE gesamt (%)
Neuronales Netz mit Rücklaufftemperatur			
Testdaten	0,067	0,004	0,167
Zehn Simulationstage basierend auf Tag 1	0,076	0,005	0,080
Zehn Simulationstage basierend auf Tag 2	0,135	0,009	0,523
Neuronales Netz ohne Rücklaufftemperatur			
Testdaten	0,084	0,004	0,181
Zehn Simulationstage basierend auf Tag 1	0,196	0,011	0,492
Zehn Simulationstage basierend auf Tag 2	0,124	0,008	0,961

Die Tabellen A.27 und A.28 zeigen die maximalen absoluten und relativen Abweichungen aller Ausgangsgrößen verglichen mit *SIR 3S*[®] für die neuronalen Netze mit und ohne Rücklaufftemperatur. Wie bereits für die vorherigen Fernwärmenetze liefern die neuronalen Netze für den Wintertag die besten Ergebnisse, während die Genauigkeit für den wärmeren Tag abnimmt.

Tabelle A.27.: Maximale absolute und relative Abweichungen der neuronalen Netze zu *SIR 3S*[®] für Netz 4 und zehn Simulationstage basierend auf Beispieltag 1 im Februar.

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	<i>MAPE</i> mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	<i>MAPE</i> ohne T_{RL} (%)
Speicherinhalt (%)	245,973	inf	inf	172,667	inf	inf
P HKW (MW)	0,948	0,836	0,104	0,670	0,536	0,059
\dot{m} Speicheranlage TG 3 (t/h)	5,494	0,904	0,099	68,649	13,851	1,322
\dot{m} Speicheranlage TG 1 (t/h)	6,137	3,999	0,311	22,127	10,565	1,320
\dot{m} Speicheranlage TG 2 (t/h)	4,033	0,669	0,058	48,684	9,655	1,566
\dot{m} Ventil Speicheranlage (t/h)	10,873	1,188	0,121	62,703	6,294	0,834
\dot{m} HW (t/h)	5,183	inf	inf	27,745	inf	inf
\dot{m} HKW (t/h)	14,951	0,784	0,099	115,028	5,627	0,831
\dot{m} DEA 2 (t/h)	0,709	0,921	0,096	8,675	13,753	1,328
\dot{m} DEA 1 (t/h)	1,377	0,900	0,099	17,304	13,874	1,322
\dot{m} Speicherbeladung (t/h)	6,825	inf	inf	2,155	inf	inf
\dot{m} Speicherentladung (t/h)	5,697	inf	inf	12,193	inf	inf
n_{Pumpe} RL DEA 2 (min^{-1})	2,243	0,264	0,038	33,770	4,012	0,420
n_{Pumpe} RL DEA 1 (min^{-1})	4,689	0,489	0,057	61,562	7,399	0,725
n_{Pumpe} RL TG 3 (min^{-1})	35,196	3,058	0,037	40,867	3,548	0,264
n_{Pumpe} RL TG 1 (min^{-1})	3,522	0,345	0,032	39,269	4,137	0,722
n_{Pumpe} VL TG 2 (min^{-1})	692,237	97,928	0,936	713,513	98,341	1,811
n_{Pumpe} RL HKW (min^{-1})	6,433	0,660	0,083	50,689	4,870	0,554
n_{Pumpe} RL HW (min^{-1})	85,189	inf	inf	97,001	inf	inf
P_{Pumpe} RL DEA 2 (kW)	0,014	1,011	0,137	0,199	17,931	1,562
P_{Pumpe} RL DEA 1 (kW)	0,107	1,609	0,205	1,482	32,430	2,368
P_{Pumpe} RL TG 3 (kW)	21,862	17,331	0,180	37,476	29,708	1,253
P_{Pumpe} RL TG 1 (kW)	1,539	1,747	0,112	10,213	14,490	2,327
P_{Pumpe} VL TG 2 (kW)	49,798	107,790	1,204	51,561	103,729	4,116
P_{Pumpe} RL HKW (kW)	323,700	73,919	10,456	431,986	90,071	9,895
P_{Pumpe} RL HW (kW)	4,579	inf	inf	9,518	inf	inf
Fehlmassenstrom (t/h)	0,154	inf	3323,994	0,137	inf	3016,071
Max. v (m/s)	0,082	4,446	0,475	0,139	7,847	1,106
Min. Δp TG 3 (bar)	0,037	0,994	0,177	0,197	4,917	0,475
Min. Δp TG 1 (bar)	0,020	0,989	0,103	0,035	1,642	0,219
Min. Δp TG 2 (bar)	0,030	1,065	0,087	0,100	3,475	0,618
Max. p_{RL} SP TG 1 (bar)	0,053	0,597	0,095	0,181	2,118	0,257
Max. p_{RL} SP TG 2 (bar)	0,009	0,124	0,016	0,012	0,156	0,015
Min. p_{RL} TG 3 (bar)	0,008	0,343	0,059	0,036	1,499	0,166
Min. p_{RL} TG 1 (bar)	0,055	0,699	0,153	0,281	3,817	0,364
Min. p_{RL} TG 2 (bar)	0,014	0,246	0,028	0,091	1,615	0,301
Max. p_{VL} TG 3 (bar)	0,049	0,554	0,070	0,059	0,673	0,108
Max. p_{VL} TG 1 (bar)	0,070	0,411	0,054	0,137	0,807	0,078
Max. p_{VL} TG 2 (bar)	0,049	0,374	0,052	0,028	0,216	0,036
Min. p_{VL} SP DEA 2 (bar)	0,039	0,578	0,112	0,059	0,863	0,139
Min. p_{VL} SP DEA 1 (bar)	0,036	0,505	0,090	0,166	2,285	0,247
Min. p_{VL} SP TG 3 (bar)	0,039	0,520	0,088	0,049	0,658	0,102

Tabelle A.27 fortgesetzt.

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	$MAPE$ mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	$MAPE$ ohne T_{RL} (%)
Min. p_{VL} SP TG 1 (bar)	0,016	0,145	0,013	0,010	0,088	0,007
Min. p_{VL} SP TG 2 (bar)	0,033	0,311	0,013	0,009	0,082	0,009
Min. T_{VL} TG 3 (°C)	0,534	0,469	0,060	2,461	2,431	0,195
Min. T_{VL} TG 1 (°C)	1,755	1,603	0,155	1,857	1,757	0,196
Min. T_{VL} TG 2 (°C)	0,469	0,385	0,029	0,814	0,671	0,065
Wärmeverbrauch TG 3 (MW)	0,265	0,733	0,066	0,238	0,603	0,061
Wärmeverbrauch TG 1 (MW)	0,214	0,329	0,056	0,387	0,773	0,067
Wärmeverbrauch TG 2 (MW)	0,179	0,522	0,069	0,337	0,945	0,061
Wärmeverbrauch gesamt (MW)	1,003	0,666	0,067	0,942	0,761	0,051
Wärmeverlust gesamt (MW)	0,014	0,178	0,022	0,167	2,128	0,341
Wärmeverlust TG 3 (MW)	0,008	0,396	0,050	0,119	6,344	0,659
Wärmeverlust TG 1 (MW)	0,006	0,310	0,027	0,057	3,285	0,532
Wärmeverlust TG 2 (MW)	0,005	0,120	0,025	0,078	2,036	0,426
\dot{m} Speicheranlage TG 1 (t/h)	5,399	0,651	0,072	61,184	7,279	1,413
\dot{m} Speicheranlage TG 2 (t/h)	4,084	0,678	0,057	48,671	9,655	1,566
\dot{m} Speicheranlage TG 3 (t/h)	5,394	0,888	0,095	68,788	13,856	1,322

Tabelle A.27 zeigt, dass das neuronale Netz mit Rücklauf­temperatur am ersten Beispieltag für fast alle Ausgangsgrößen sowohl absolut als auch relativ sehr niedrige Abweichungen zu $SIR\ 3S^{\circ}$ aufweist. Die grau markierten Felder zeigen relative Abweichungen größer 3%, welche aber nur bei sehr niedrigen absoluten Abweichungen um 0 auftreten. Der erste Ausgang, der Speicherinhalt, kann wie bereits beschrieben nicht gut vorausgesagt werden. Hier sollte statt einer direkten Vorhersage des Füllstandes eine Vorhersage der Füllstands­differenz pro Zeitschritt getestet werden. Ebenfalls hohe absolute Abweichungen zeigt die Drehzahl der Vorlaufpumpen in Teilgebiet 2. Im Durchschnitt sind die Abweichungen hier mit unter einem Prozent sehr gering, allerdings entstehen Ausreißer. Die Pumpendrehzahlen (n_{Pumpe}) befinden sich in allen simulierten Fällen im mittleren Bereich zwischen Minimum und Maximum. Somit stellt auch eine stark falsche Vorhersage hier kein direktes Problem für eine spätere Optimierung dar. Dennoch kann die Vorhersage noch verbessert werden. Auch für die Pumpendrehzahl des HW entstehen größere absolute Fehler, welche allerdings bei allgemein sehr hohen Drehzahlen entstehen. Für einige Pumpenleistungen (n_{Pumpe}) berechnet das neuronale Netz zudem Ausreißer und hohe Abweichungen können beobachtet werden. Die Pumpenleistungen werden in der späteren Optimierung nicht berücksichtigt, können aber in einer weiterführenden Optimierung unter Betrachtung der Gesamtleistung des Systems, wie für Fernwärmenetz 1 vorgestellt, zum Einsatz kommen. In diesem Fall sollte die Berechnung der Pumpenleistung nachgebessert werden. Ansonsten kann das neuronale Netz mit Rücklauf­temperatur Beispieltag 1 sehr gut abbilden. Auch das Netz ohne Rücklauf­temperatur zeigt im Durchschnitt niedrige relative Abweichungen, wobei die maximalen Abweichungen hier für mehrere Ausgänge deutlich höher liegen.

Für den Maitag zeigt Tabelle A.28 eine Zunahme der Abweichungen. Für viele Ausgänge steigt nur die maximale relative Abweichung an, während durchschnittliche relative und absolute Abweichungen niedrig sind. Problematisch sind wie für die Vorhersage an Beispieltag 1 die Vorhersage des Speicherinhalts sowie einiger Pumpendrehzahlen und -leistungen. Für eine spätere Optimierung ist besonders kritisch, dass verglichen zum Wintertag noch höhere Abweichungen der Drehzahlen auftreten, welche teilweise Restriktionen verletzen. Somit sollte das Modell hier verbessert werden.

Tabelle A.28.: Maximale absolute und relative Abweichungen der neuronalen Netze zu $SIR\ 3S^{\circledR}$ für Netz 4 und zehn Simulationstage basierend auf Beispieltag 2 im Mai.

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	$MAPE$ mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	$MAPE$ ohne T_{RL} (%)
Speicherinhalt (%)	45,156	inf	inf	33,029	inf	inf
P HKW (MW)	0,812	2,168	0,699	1,169	3,718	0,841
\dot{m} Speicheranlage TG 3 (t/h)	9,978	2,840	0,787	56,695	14,491	1,936
\dot{m} Speicheranlage TG 1 (t/h)	16,886	1637,667	12,228	16,222	4561,956	14,852
\dot{m} Speicheranlage TG 2 (t/h)	8,138	5,622	0,591	45,181	25,207	2,481
\dot{m} Ventil Speicheranlage (t/h)	12,829	3,261	0,566	22,298	5,806	1,415
\dot{m} HW (t/h)	8,223	inf	inf	29,001	inf	inf
\dot{m} HKW (t/h)	17,028	2,039	0,349	64,571	7,015	1,326
\dot{m} DEA 2 (t/h)	1,173	2,467	0,622	7,169	14,211	1,926
\dot{m} DEA 1 (t/h)	2,485	2,720	0,731	14,281	14,511	1,953
\dot{m} Speicherbeladung (t/h)	16,205	inf	inf	6,358	inf	inf
\dot{m} Speicherentladung (t/h)	8,192	inf	inf	11,084	inf	inf
n_{Pumpe} RL DEA 2 (min^{-1})	3,391	0,427	0,104	21,365	2,711	0,339
n_{Pumpe} RL DEA 1 (min^{-1})	8,017	1,046	0,310	39,849	5,400	0,688
n_{Pumpe} RL TG 3 (min^{-1})	43,134	3,571	1,755	219,890	18,664	1,746
n_{Pumpe} RL TG 1 (min^{-1})	109,493	13,050	0,581	77,656	9,256	0,711
n_{Pumpe} VL TG 2 (min^{-1})	734,186	4557,306	13,583	673,796	2624,044	8,024
n_{Pumpe} RL HKW (min^{-1})	8,861	1,334	0,311	13,664	2,088	0,470
n_{Pumpe} RL HW (min^{-1})	140,883	inf	inf	61,471	inf	inf
P_{Pumpe} RL DEA 2 (kW)	0,024	2,499	0,648	0,110	12,153	1,497
P_{Pumpe} RL DEA 1 (kW)	0,210	5,714	1,166	0,634	19,160	2,432
P_{Pumpe} RL TG 3 (kW)	28,698	22,766	6,590	122,039	149,237	9,074
P_{Pumpe} RL TG 1 (kW)	5,823	14,034	2,245	8,427	41,878	4,024
P_{Pumpe} VL TG 2 (kW)	39,120	33647,490	75,755	28,586	45765,725	136,716
P_{Pumpe} RL HKW (kW)	945,269	676,729	194,924	345,686	293,680	71,930
P_{Pumpe} RL HW (kW)	3,651	inf	inf	7,044	inf	inf
Fehlmassenstrom (t/h)	0,172	inf	inf	0,182	inf	inf
Max. v (m/s)	0,158	13,436	3,124	0,398	33,729	9,751
Min. Δp TG 3 (bar)	0,091	2,205	0,511	0,099	2,423	0,247
Min. Δp TG 1 (bar)	0,034	1,797	0,224	0,041	2,149	0,242
Min. Δp TG 2 (bar)	0,020	0,556	0,086	0,067	1,898	0,245
Max. p_{RL} SP TG 1 (bar)	0,100	1,022	0,285	0,089	0,909	0,242
Max. p_{RL} SP TG 2 (bar)	0,004	0,060	0,006	0,007	0,098	0,008
Min. p_{RL} TG 3 (bar)	0,013	0,550	0,133	0,017	0,704	0,072
Min. p_{RL} TG 1 (bar)	0,119	1,316	0,400	0,084	0,952	0,293
Min. p_{RL} TG 2 (bar)	0,017	0,274	0,040	0,068	1,091	0,130
Max. p_{VL} TG 3 (bar)	0,037	0,433	0,091	0,086	1,005	0,093
Max. p_{VL} TG 1 (bar)	0,045	0,268	0,058	0,041	0,244	0,063
Max. p_{VL} TG 2 (bar)	0,017	0,132	0,027	0,016	0,118	0,030
Min. p_{VL} SP DEA 2 (bar)	0,038	0,550	0,102	0,060	0,873	0,096

Tabelle A.28 fortgesetzt.

Ausgangsgröße	Max. abs. mit T_{RL}	Max. rel. mit T_{RL} (%)	<i>MAPE</i> mit T_{RL} (%)	Max. abs. ohne T_{RL}	Max. rel. ohne T_{RL} (%)	<i>MAPE</i> ohne T_{RL} (%)
Min. p_{VL} SP DEA 1 (bar)	0,029	0,393	0,098	0,072	0,960	0,145
Min. p_{VL} SP TG 3 (bar)	0,036	0,479	0,092	0,071	0,944	0,075
Min. p_{VL} SP TG 1 (bar)	0,007	0,068	0,005	0,005	0,046	0,007
Min. p_{VL} SP TG 2 (bar)	0,005	0,047	0,005	0,002	0,021	0,005
Min. T_{VL} TG 3 (°C)	1,145	1,429	0,207	2,582	3,180	0,539
Min. T_{VL} TG 1 (°C)	3,934	5,135	1,354	3,256	4,022	1,134
Min. T_{VL} TG 2 (°C)	1,139	1,349	0,249	0,906	1,111	0,220
Wärmeverbrauch TG 3 (MW)	0,748	7,561	1,803	0,464	4,171	0,779
Wärmeverbrauch TG 1 (MW)	0,718	6,389	1,031	0,492	4,115	0,538
Wärmeverbrauch TG 2 (MW)	0,593	12,985	1,923	0,195	5,203	0,811
Wärmeverbrauch gesamt (MW)	1,095	4,538	0,933	0,768	2,463	0,497
Wärmeverlust gesamt (MW)	0,025	0,362	0,079	0,164	2,580	0,385
Wärmeverlust TG 3 (MW)	0,014	0,831	0,163	0,102	6,181	0,723
Wärmeverlust TG 1 (MW)	0,006	0,490	0,124	0,059	4,437	0,585
Wärmeverlust TG 2 (MW)	0,017	0,558	0,085	0,108	3,308	0,538
\dot{m} Speicheranlage TG 1 (t/h)	11,530	2,782	0,551	63,335	13,207	2,595
\dot{m} Speicheranlage TG 2 (t/h)	7,860	5,425	0,598	45,344	25,281	2,492
\dot{m} Speicheranlage TG 3 (t/h)	9,527	2,803	0,709	56,509	14,444	1,931

A.2. Optimierung von Fernwärmenetz 4

In Fernwärmenetz 4 werden die Leistungen und Vorlauftemperaturen des HKW und des HW optimiert. Die Druckhaltung wird bisher konstant belassen. Die Speicher werden fix entsprechend der wirtschaftlichen Optimierung be- oder entladen. Hier findet keine weitere Optimierung statt. Vor- und Rücklauftemperaturen der Speicher werden noch nicht prognostiziert und nur nach Messdaten der Beispieltage verwendet. Auch der Massenstrom der Beimischung wird nicht angepasst und muss entsprechend den Beispieltagen vorgegeben oder angenommen werden. Wie für Fernwärmenetz 3 gibt es in Netz 4 mehrere Teilgebiete, für welche die Netzsimulation einzelne Einspeiseleistungen benötigt. Allerdings findet nur eine Prognose der Gesamtleistung und nicht der drei Teilgebietsleistungen statt. Somit müssen Annahmen der Wärmeverteilung getroffen werden. Für die Optimierung der Beispieltage wird die Verteilung dieser Tage festgelegt. Die folgenden Restriktionen werden angewandt:

- Nenndrehzahl als maximale Drehzahl für alle Pumpen
- Minimale Drehzahl der DEA
- Minimale und maximale Massenströme und Leistungen des HKW, des HW und der Speicheranlage
- Minimaler und maximaler Speicherinhalt. Diese Restriktion kann noch nicht berücksichtigt werden, da die Vorhersage des Speicherinhalts unzureichend ist.
- Netzweite maximale Strömungsgeschwindigkeit
- Minimaler Differenzdruck in den drei Teilgebieten
- Minimale Vorlauftemperatur von 90 °C an allen Kundenanlagen
- Minimaler Rücklaufdruck der drei Teilgebiete und maximaler Rücklaufdruck an den Schlechtpunkten von Teilgebiet 1 und 2
- Minimaler und maximaler Vorlaufdruck der drei Teilgebiete
- Maximale Netzeinspeisetemperaturgradienten am Ausgang der Stationen
- Außentemperaturabhängige Vorlauftemperaturgrenzen für die Wärmeerzeugungsanlagen entsprechend der wirtschaftlichen Optimierung
- Maximaler Fehlmassenstrom von 10 t/h

Die hier verwendeten Restriktionen basieren in Teilen auf Angaben des Netzbetreibers und wurden um eigene Annahmen ergänzt, um eine vollständige Optimierung zu ermöglichen. Wie bei den vorherigen Netzen besteht jederzeit die Möglichkeit, die Restriktionen, das *SIR 3S*[®] Modell oder die neuronalen Netze anzupassen.

Optimierung nach Leistungsprognose und wirtschaftlichen Sollwerten

Nun wird eine 24-Stunden-Optimierung von Beispieltag 1 mit Sollwerten der wirtschaftlichen Optimierung sowie deren Leistungsprognose getestet. Verwendet wird das neuronale Netz für stationäre Simulationen aus Kapitel A.21 mit der Rücklauftemperatur als Eingang. Diese wird dabei fix entsprechend den Messwerten an Beispieltag 1 vorgegeben. Als Optimierungsalgorithmen kommen der lokale Ansatz SLSQP und der globale Ansatz *differential evolution* zum Einsatz. Für die Optimierung stehen hier drei Variablen zur Verfügung. Diese sind die Leistung des HW und die Vorlauftemperaturen des HKW und HW. Die Leistung des HKW ergibt sich aus der Leistungsbilanz mit dem Heizwerk, der

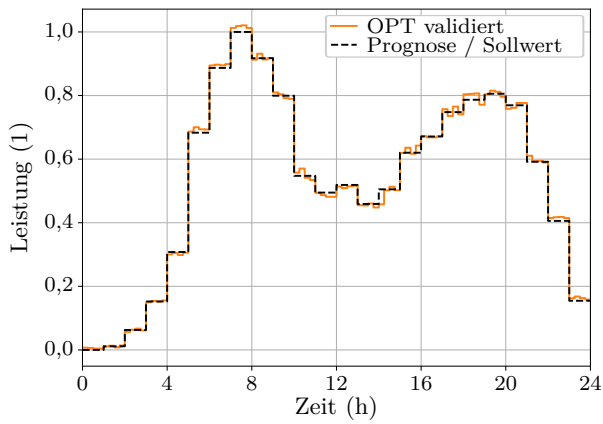
Speicheranlage und der gesamt einzuspeisenden Leistung, welche entsprechend der Leistungsprognose fix ist.

Als Zielfunktion wird die folgende Gleichung (A.1) verwendet, welche die Abweichungen der Erzeugerleistungen und Vorlauftemperaturen von den wirtschaftlich optimierten Sollwerten mit einbezieht. Die beiden Komponenten der Zielfunktion sind mit Gewichtungsfaktoren versehen. Das HKW erhält mit 100 jeweils die höhere Gewichtung, da dieses auch für die Stromproduktion zum Einsatz kommt. Bei Bedarf ist eine Änderung der Gewichtung oder Komponenten der Funktion möglich, um andere Optimierungsziele zu erreichen.

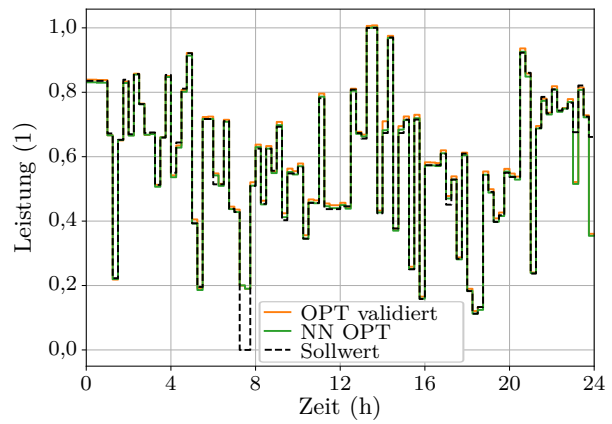
$$\begin{aligned}
\Delta P_{HKW} &= 100 \cdot (P_{HKW} - SW_{P_{HKW}}) \\
\Delta P_{HW} &= 1 \cdot (P_{HW} - SW_{P_{HW}}) \\
\Delta T_{VL, HKW} &= 100 \cdot (T_{VL, HKW} - SW_{T_{VL, HKW}}) \\
\Delta T_{VL, HW} &= 1 \cdot (T_{VL, HW} - SW_{T_{VL, HW}}) \\
f &= 0,5 \cdot (\Delta P_{HKW}^2 + \Delta P_{HW}^2 + \Delta T_{VL, HWK}^2 + \Delta T_{VL, HW}^2) \quad (A.1)
\end{aligned}$$

Die Abbildungen A.2a und A.2b zeigen das Ergebnis für die gesamte Einspeiseleistung und die des HKW. Beide Leistungen folgen dem Sollwert. Gesamt wird zu einigen Zeitpunkten eine leicht zu hohe oder niedrige Leistung eingespeist. Die maximale Abweichung beträgt etwa 0,85 MW. Gegen 8 Uhr kann die niedrige, von der wirtschaftlichen Optimierung vorgegebene Leistung des HKW nicht eingehalten werden. Am Ende des Tages wird an zwei Zeitpunkten weniger Leistung eingespeist als gefordert. Die Leistung des HW zeigt Abbildung A.2c. Größtenteils können die Sollwerte des HW erreicht werden. Zu den Zeitpunkten gegen 8 Uhr und am Ende des Tages wurden auch Abweichungen vom Sollwert des HKW beobachtet. Für die Abweichung am Morgen kann dies in Zusammenhang mit dem minimalen Differenzdruck in Teilgebiet 1 gebracht werden, welcher in der Optimierung bis auf den Minimalwert von 0 bar (mit Versatz) fällt und in Abbildung A.2d dargestellt ist. Die geforderte niedrige Leistung des HKW würde also zu nicht akzeptablen niedrigen Differenzdrücken führen, weshalb vom HKW mehr und vom HW weniger Leistung eingespeist wird. In Abbildung A.2d ist auch zu erkennen, dass der minimale Differenzdruck in der *SIR 3S*[®] Validierungsrechnung sogar unter den Grenzwert von 0,0 fällt. Zwischen neuronalem Netz und *SIR 3S*[®] entsteht bei dieser Fahrweise also ein Berechnungsfehler. Hier sollte die Optimierung im derzeitigen Zustand mit robusteren Grenzwerten betrieben werden, um zu niedrige Differenzdrücke zu vermeiden. Allgemein können auch die neuronalen Netze noch verbessert werden. Die Be- und Entladeleistung der Speicheranlage sind in den Abbildungen A.2e und A.2f dargestellt. Wie vorgegeben, folgen die Leistungen den Sollwerten. Zu einigen Zeitpunkten wird allerdings nicht die geforderte Beladeleistung erreicht. Die maximalen Abweichungen liegen unter 1,5 MW. Die Entladeleistung der Speicheranlage wird hingegen sehr genau getroffen. Abbildung A.3 zeigt des Weiteren die optimierten Vorlauftemperaturen. Die Vorlauftemperatur des HKW folgt sehr genau dem Sollwert, während die Vorlauftemperatur des HW stärker abweicht. Dieses Verhalten wird durch die Wahl der Gewichtungsfaktoren in der Zielfunktion der Optimierung begünstigt.

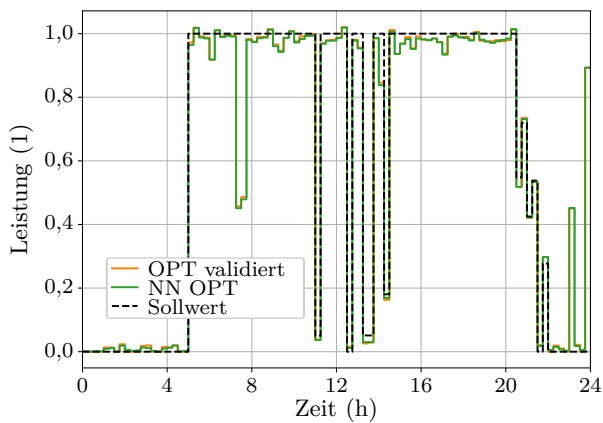
Mit dem neuronalen Netz ohne Rücklauftemperatur als Eingang ist eine Optimierung ebenfalls möglich. Die Abweichungen zwischen den Optimierungsergebnissen des neuronalen Netzes und der Validierungssimulation in *SIR 3S*[®] sind jedoch höher. Die eingespeiste Gesamtleistung würde so um bis zu 10% abweichen und die Optimierungsergebnisse sind somit nicht ausreichend genau. Eine Rücklauftemperaturvorhersage wurde für Netz 4 bisher nicht umgesetzt. Somit ist auch keine Optimierung vorhanden, welche diese enthält. Auch hier kann aber eine Vorhersage mittels neuronaler Netze basierend auf Messdaten nachträglich implementiert werden. Für den zweiten Beispieltag im Mai ist zwar die Optimierung erfolgreich, die Validierungsrechnung in *SIR 3S*[®] liefert allerdings keine plausiblen Ergebnisse. So kann das neuronale Netz hier noch nicht für eine Optimierung verwendet werden.



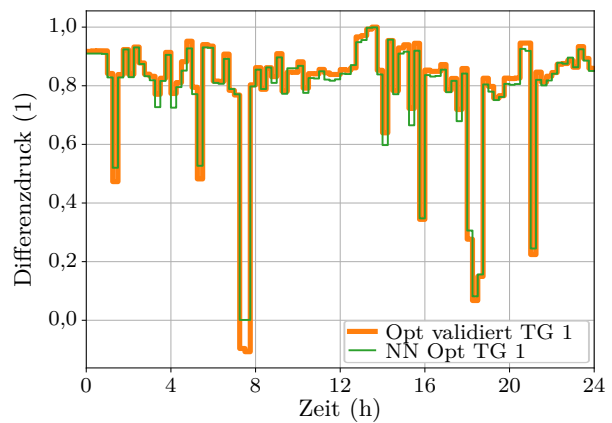
a) Gesamtleistung



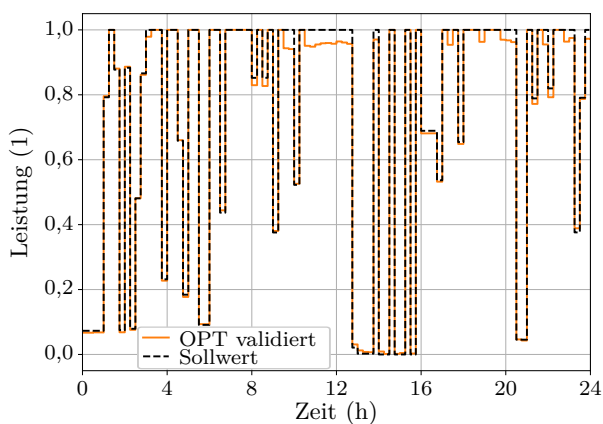
b) Leistung des HKW



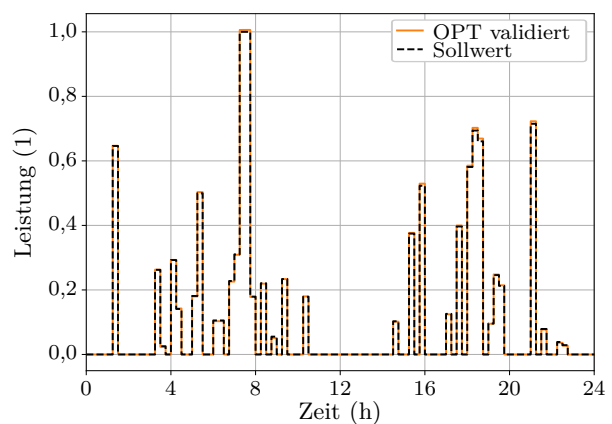
c) Leistung des HW



d) Min. Differenzdruck in Teilgebiet (TG) 1



e) Beladeleistung der Speicheranlage



f) Entladeleistung der Speicheranlage

Abbildung A.2.: Normierte Simulationsergebnisse für Beispieltag 1 im Februar nach stationärer Optimierung von Fernwärmenetz 4.

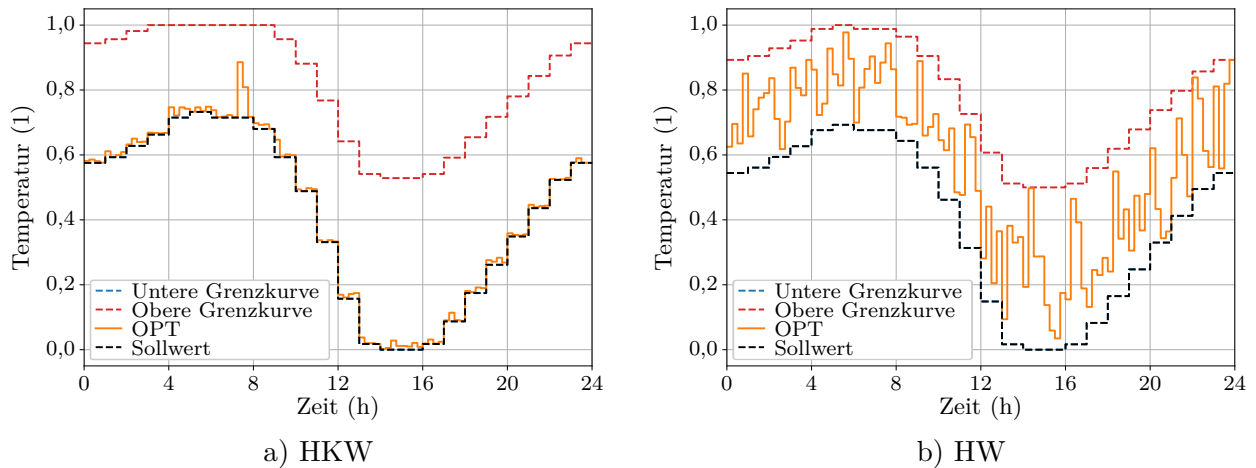


Abbildung A.3.: Normierte Vorlauftemperaturen nach stationärer Optimierung von Fernwärmenetz 4 an Beispieltag 1 im Februar.

Die folgende Tabelle A.29 zeigt die Dauer der Optimierungen für die zwei Beispieltage mit den neuronalen Netzen mit und ohne Rücklauftemperatur. Die Optimierung mit dem lokalen Ansatz SLSQP ist in fast allen Optimierungsschritten möglich. In einigen Fällen wird allerdings keine Lösung innerhalb der maximal angesetzten 1000 Iterationen oder keine geeignete Suchrichtung gefunden, in welcher die Zielfunktion abnimmt. Deshalb dauert diese Optimierung auch länger als mit dem globalen Ansatz *differential evolution* und liefert für einige Zeitpunkte kein Betriebsoptimum. Insgesamt ist eine Optimierung mit *differential evolution* für den Einsatz in einer durch Personal zu validierenden Optimierung schnell genug. Für einen Einsatz in einer MPC kann der lokale Ansatz im Detail weiter untersucht werden. Zudem müssen neuronale Netze mindestens für Beispieltag 2 im Frühling und auch für Sommertage, für welche bisher keine Validierung durchgeführt wurde, nachgebessert werden. Da nur die neuronalen Netze mit Rücklauftemperatur gute Optimierungsergebnisse liefern, ist eine Rücklauftemperaturvorhersage für Fernwärmenetz 4 notwendig. Wichtig ist auch, dass die Optimierungsergebnisse bisher nur mit stationären Validierungsrechnungen in *SIR 3S*[®] überprüft wurden. Eine Nachrechnung mit instationären Simulationen hat zwar für Fernwärmenetz 2 und 3 keine prinzipiellen Probleme gezeigt, sollte hier aber dennoch durchgeführt werden.

Tabelle A.29.: Optimierungsdauer für Netz 4 an den zwei Beispieltagen mit Leistungsvorgabe aus der Vortags-Prognose.

	Dauer mit SLSQP (s)	Dauer mit <i>differential evolution</i> (s)
Tag 1 (Februar) mit T_{RL} Vorgabe	8,22	3,74
Tag 1 (Februar) ohne T_{RL} Vorgabe	0,80	3,50
Tag 2 (Mai) mit T_{RL} Vorgabe	X	X
Tag 2 (Mai) ohne T_{RL} Vorgabe	X	X

Optimierung mit Einspeiseleistung aus Prozessdaten und wirtschaftlichen Sollwerten

Für eine Untersuchung der Optimierung von Fernwärmenetz 4 bei abweichenden Leistungsprognosen werden wie bei den vorherigen Netzen Prozessdaten von den Beispieltagen für die Vorgabe der gesamten Wärmeleistung und die wirtschaftlich optimierten Leistungs- und Vorlauftemperatursollwerte vom Vortrag verwendet. Die gleichen Restriktionen wie zuvor kommen zum Einsatz. Eine Optimierung ist in diesem Fall für beide Beispieltage möglich. Die tatsächlich an Beispieltag 2 gemessene Leistung lag, wie in Abbildung A.4a dargestellt, deutlich höher als die für die vorherige Optimierung verwendete

Leistungsprognose. Daher befinden sich die Eingänge der neuronalen Netze hier innerhalb der verwendeten Trainingsdaten und Tagesverläufe können genau berechnet werden. Abbildung A.4b zeigt das Ergebnis für die Leistung des HKW, welche bis auf einige Zeitpunkte genau erreicht wird. Die Differenz zwischen Prognose und tatsächlicher Einspeiseleistung wird hauptsächlich vom HW ausgeglichen, wie in Abbildung A.4c sichtbar. Für die Vorlauftemperatur des HKW wird der Sollwert in Abbildung A.4d, welcher auf der unteren Grenzkurve liegt, gut getroffen.

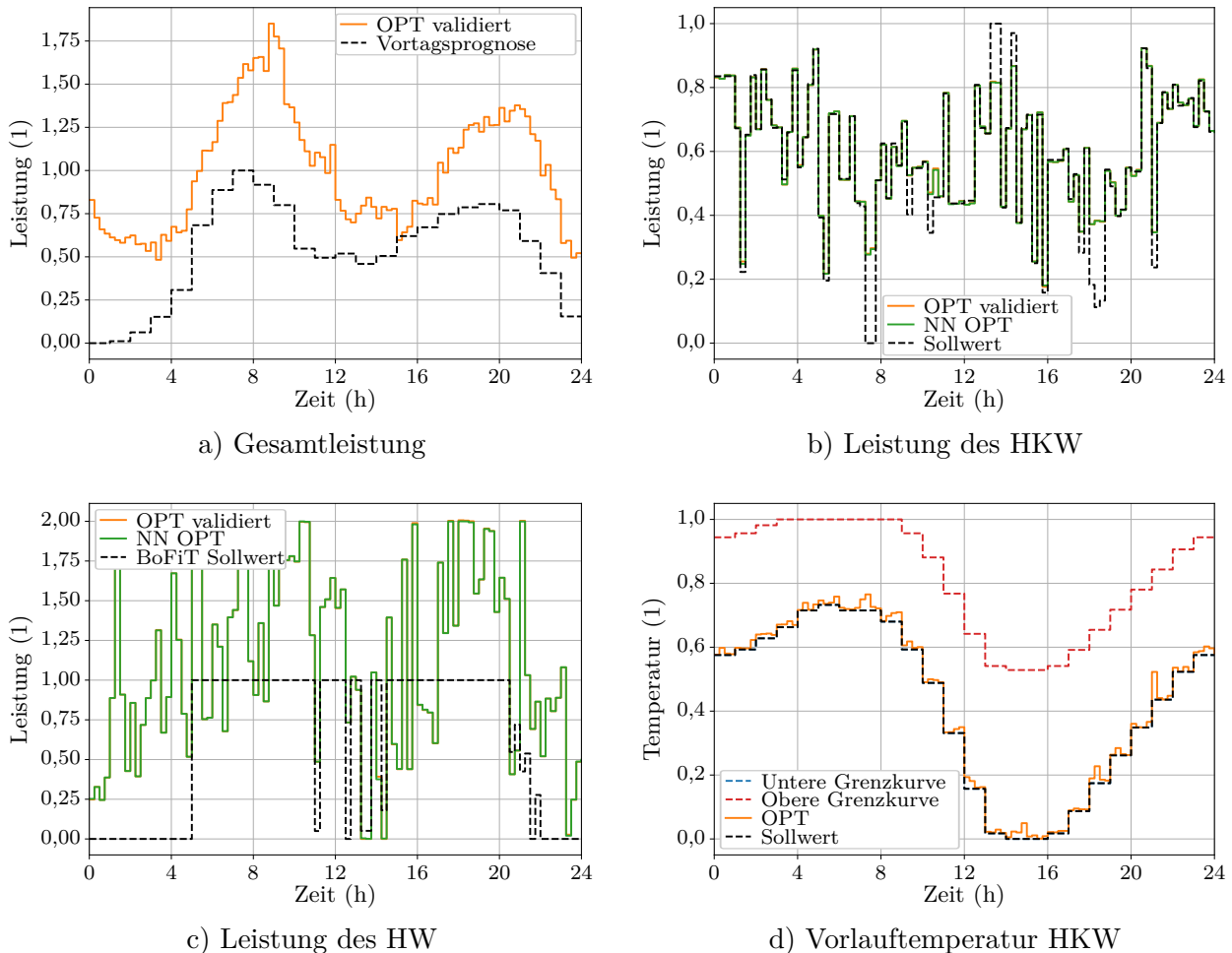


Abbildung A.4.: Normierte Simulationsergebnisse nach stationärer Optimierung von Fernwärmenetz 4 an Beispieltag 2 im Mai mit Leistungsvorgabe aus Prozessdaten.

In diesem Fall ist die Optimierung sowohl mit SLSQP als auch mit *differential evolution* möglich. Tabelle A.30 zeigt die jeweilige Optimierungsdauer an den Beispieltagen.

Tabelle A.30.: Optimierungsdauer für Netz 4 an den zwei Beispieltagen mit Leistungsvorgabe aus Prozessdaten.

	Dauer mit SLSQP (s)	Dauer mit <i>differential evolution</i> (s)
Tag 1 (Februar) mit T_{RL} Vorgabe	0,88	2,55
Tag 1 (Februar) ohne T_{RL} Vorgabe	0,96	2,53
Tag 2 (Mai) mit T_{RL} Vorgabe	1,02	2,77
Tag 2 (Mai) ohne T_{RL} Vorgabe	0,95	2,74

Verwendung des Vorlaufs als Temperaturspeicher

An den Beispieltagen werden die Vorlaufleitungen von Fernwärmenetz 4 nicht als Speicher verwendet. Theoretisch ist dies möglich, praktisch wird aber überschüssige Leistung in der Speicheranlage gespeichert oder bei einem Überschuss der Wärmeabnahme Leistung der Anlage entnommen.

Optimierung der Wärmeverluste

Eine Optimierung der Wärmeverluste wurde für Fernwärmenetz 1 ausführlich in den Kapiteln 7.1.2 und 7.1.3 diskutiert. Auch für Fernwärmenetz 2 und 3 wurde sie untersucht, wobei für diese Netze vor allem zusätzliche Restriktionen in Form von Gradienten der Heizwerkleistungen notwendig wären. Zudem weicht die Leistung des HKW stark vom Sollwert ab. Dies ist auch für Fernwärmenetz 4 zu erwarten. Hier wurde noch keine Wärmeverlustoptimierung durchgeführt. Prinzipiell ist dies aber durch eine Anpassung der Zielfunktion der Optimierung möglich.

B. Symbolverzeichnis

A Fläche.

A_{inz} Knoten-Kanten-Inzidenzmatrix.

B_{inz} Maschen-Kanten-Inzidenzmatrix.

c_p Spezifische Wärmekapazität.

d Rohrdurchmesser.

η Wirkungsgrad.

h Höhe.

k Rohrrauigkeit.

l Rohrlänge.

λ Rohrreibungsbeiwert.

λ_{Bo} Wärmeleitfähigkeit des Erdreichs.

λ_d Wärmeleitfähigkeit des Dämmmaterials.

\dot{m} Massenstrom.

μ Viskosität.

n Drehzahl.

P Leistung.

p Druck.

\dot{Q} Wärmestrom.

R^2 Bestimmtheitsmaß.

r Radius.

Re Reynoldszahl.

ρ Dichte.

R Rohrleitungswiderstand.

T Temperatur.

t Zeit.

T_A Rohrausgangstemperatur.

T_E Rohreingangstemperatur.

T_{RL} Rücklaufstemperatur.

T_U Umgebungstemperatur.

T_{VL} Vorlaufstemperatur.

U Umfang.

U_R Wärmedurchgangskoeffizient.

\dot{V} Volumenstrom.

v Strömungsgeschwindigkeit.

ζ Zusätzlicher Widerstandsbeiwert eines Rohres.

C. Abkürzungsverzeichnis

Adam *Adaptive Moment Estimation.*

ARIMAX *AutoRegressive Integrated Moving Average with eXogenous terms.*

ARIX *AutoRegressive Integrated with eXogenous terms.*

ARMAX *AutoRegressive-Moving Average with eXogenous terms.*

ARX *AutoRegressive with eXogenous terms.*

Bagging *Bootstrap-Aggregating.*

DEA *Druckerhöhungsanlage.*

GIS *Geoinformationssystem.*

HKW *Heizkraftwerk.*

HW *Heizwerk.*

KWK *Kraft-Wärme-Kopplung.*

LP *Linear Programming.*

LTI *Linear Time Invariant.*

MAE *Mean Absolute Error.*

MAPE *Mean Absolute Percentage Error.*

MILP *Mixed-Integer Linear Programming.*

MINLP *Mixed-Integer Non-Linear Programming.*

MPC *Model Predictive Control.*

MSE *Mean Squared Error.*

nlarx *NonLinear AutoRegressive with eXogenous input.*

NN *Neuronales Netz.*

NRMSE *Normalized Root Mean Squared Error.*

ONNX *Open Neural Network Exchange.*

QP *Quadratic Programming.*

ReLU *Rectifier Linear Unit.*

RMSE *Root Mean Squared Error.*

RNN *Recurrent Neural Network.*

RL *Rücklauf.*

SGD *Stochastic Gradient Descent.*

shgo *simplicial homology global optimization.*

SLSQP *Sequential Least Squares Programming.*

SP *Schlechtpunkt.*

SQP *Sequential Quadratic Programming.*

TG *Teilgebiet.*

UFS *Umformstation.*

VL *Vorlauf.*

WORHP *We Optimize Really Huge Problems.*

XGBoost *eXtreme Gradient Boosting.*

D. Abbildungsverzeichnis

1.1.	Beispiel eines Fernwärmenetzes mit zentraler und dezentraler Wärmeversorgung. . . .	1
2.1.	Schematischer Aufbau eines Fernwärmenetzes nach [100].	5
2.2.	Beispielhaftes Druckdiagramm eines Fernwärmenetzes nach [92, S. 59].	6
2.3.	Beispiele für Topologien von Fernwärmenetzen.	8
2.4.	Beispiel für ein maschenförmiges Fernwärmenetz.	9
2.5.	Schematische Darstellung einer Rohrleitung mit Mediumrohr, Wärmedämmung und Mantelrohr nach [100, S. 78].	9
2.6.	Wärmetauscher nach dem Gleich- und Gegenstromprinzip nach [100].	10
2.7.	Konvexe Funktion mit markiertem globalem Minimum und nicht-konvexe Funktion mit einem globalen (1) sowie einem lokalen (2) Minimum, nach [95, S. 15].	12
2.8.	Prinzip der Modellprädiktiven Regelung nach [32, S. 39].	20
2.9.	Struktur eines nicht-linearen ARX Modells nach [123].	24
2.10.	Binärer Entscheidungsbaum nach [42, S. 129].	29
2.11.	Modell eines Neurons nach [42, S. 169].	30
2.12.	Neuronales Netz mit einem <i>Hidden-Layer</i> nach [42, S. 179] [72, S. 44].	30
2.13.	Verschiedene Aktivierungsfunktionen für die Neuronen der neuronalen Netze.	31
3.1.	Jahresdauerlinie eines Fernwärmenetzes nach [92, S. 18].	39
3.2.	Jahresdauerlinie eines Fernwärmenetzes mit Verteilung der Wärmeleistung auf verschiedene Wärmeerzeugungsanlagen.	40
4.1.	Derzeitiges Regelungskonzept für Fernwärmenetze.	51
4.2.	Konzept für ein automatisiertes Fernwärmenetz unter Verwendung einer MPC.	52
4.3.	Konzept für ein halb-automatisiertes Fernwärmenetz unter Verwendung einer nicht-linearen Optimierung.	53
5.1.	Der Zusammenhang zwischen Druckdifferenz und Volumenstrom einer Pumpe dargestellt in der Pumpenkennlinie.	59
5.2.	Beispiel einer Pumpenkennlinie bei 50 % und 100 % der Drehzahl zusammen mit der Anlagenkennlinie.	60
5.3.	Schematische Darstellung von Rohrleitungen für die Erläuterung des ersten Kirchhoffschen Gesetzes, der Knotenregel.	63
5.4.	Schematische Darstellung von Rohrleitungen für die Erläuterung des zweiten Kirchhoffschen Gesetzes, der Maschenregel.	63
5.5.	Parallel verlegte Vor- und Rücklaufleitungen in Einzelrohrausführung nach [92, S. 127].	64
5.6.	Leistungsbilanz an einem Rohrelement nach [61, S. 89].	67
5.7.	Bestimmung des Eintrittszeitpunktes t_E nach [61, S. 95].	68
5.8.	Hydraulisches Modell einer Rohrleitung in <i>Simulink</i> [®]	70
5.9.	Beispiel eines Fernwärmenetzes mit einer Masche und markierten Knoten und Kanten.	75
6.1.	Vereinfachte, nicht-maßstäbliche Darstellung des realen Fernwärmenetzes 1 mit Druckerhöhungs- und Wärmeerzeugungsanlagen.	82
6.2.	Aggregiertes und reduziertes Fernwärmenetz 1.	83

6.3.	Beispiel für <i>Overfitting</i> eines neuronalen Netzes für Fernwärmenetz 1 mit unterschiedlicher Anzahl der für das Training verwendeten Datensätze.	90
6.4.	Lernkurven für die neuronalen Netze von Fernwärmenetz 1.	96
6.5.	Vergleich des neuronalen Netzes (<i>PyTorch</i>) von Fernwärmenetz 1 mit <i>SIR 3S</i> [®] . Alle Größen sind normiert dargestellt.	96
6.6.	Vergleich von normierten Simulationsergebnissen von Beispieltag 1 im Januar mit dem aggregierten und vollständigen <i>SIR 3S</i> [®] Netzmodell.	97
6.7.	Quasi-stationäre und instationäre <i>SIR 3S</i> [®] Simulation von Beispieltag 1 im Januar mit IST- und variierter Vorlauftemperatur. Alle Größen sind mit Versatz der <i>y</i> -Achse dargestellt.	99
6.8.	Normierte Ergebnisse für die Systemidentifikation der maximalen Strömungsgeschwindigkeit in Fernwärmenetz 1 mit IST und variierter Vorlauftemperatur.	100
6.9.	Für die Systemidentifikation der Wärmeeinspeisung und Pumpenleistung verwendete normierte Vorlauftemperaturen.	100
6.10.	Normierte Ergebnisse der Systemidentifikation für die Wärmeeinspeisung und Pumpenleistung.	101
6.11.	Normierte Ergebnisse der maximalen Strömungsgeschwindigkeit an Tag 1 im Januar mit IST-Vorlauftemperatur und der <i>MATLAB</i> [®] <i>Deep Learning Toolbox</i>	102
6.12.	Vereinfachte, nicht-maßstäbliche Darstellung des realen Fernwärmenetzes 2 mit Druckerhöhungs- und Wärmeerzeugungsanlagen.	103
6.13.	Lernkurven für die neuronalen Netze (NN) von Fernwärmenetz 2.	109
6.14.	Vergleich normierter Simulationsergebnisse von Netz 2 an Tag 1 im Februar mit dem <i>SIR 3S</i> [®] Modell, dem neuronalen Netz ohne (NN) und mit Rücklauftemperatur (NN mit TRL) als Eingangsvariable.	111
6.15.	Vergleich normierter Simulationsergebnisse von Netz 2 an Tag 3 im Juli mit dem <i>SIR 3S</i> [®] Modell, dem neuronalen Netz ohne (NN) und mit Rücklauftemperatur (NN mit TRL) als Eingangsvariable.	112
6.16.	Normierte Rücklauftemperatur des HKW in Netz 2. Messdaten verglichen mit der Vorhersage durch ein Neuronales Netz (NN).	114
6.17.	Vergleich normierter Simulationsergebnisse von Netz 2 an Tag 1 im Februar mit dem <i>SIR 3S</i> [®] Modell, dem neuronalen Netz ohne (NN) und mit Vorhersage der Rücklauftemperatur (NN mit TRL Vorhersage).	114
6.18.	Vereinfachte, nicht-maßstäbliche Darstellung des realen Fernwärmenetzes 3 mit Druckerhöhungs- und Wärmeerzeugungsanlagen.	115
6.19.	Lernkurven für die neuronalen Netze (NN) von Fernwärmenetz 3.	120
6.20.	Vergleich normierter Simulationsergebnisse von Netz 3 an Tag 1 im Februar mit <i>SIR 3S</i> [®] , dem neuronalen Netz (NN) ohne und mit Rücklauftemperatur (TRL).	122
6.21.	Vergleich normierter Simulationsergebnisse für die Drehzahlen der DEA Vor- und Rücklaufpumpen (VL und RL) in Netz 3 an Tag 1 im Februar mit <i>SIR 3S</i> [®] , dem neuronalen Netz (NN) ohne und mit Rücklauftemperatur (TRL).	122
6.22.	Simulierte, normierte min. Differenzdrücke in TG 1 und TG 2 von Netz 3 an Tag 1 mit <i>SIR 3S</i> [®] , dem neuronalen Netz (NN) ohne und mit Rücklauftemperatur (TRL).	123
6.23.	Vergleich normierter Simulationsergebnisse von Netz 3 an Tag 3 im Juli mit <i>SIR 3S</i> [®] , dem neuronalen Netz (NN) ohne und mit Rücklauftemperatur (TRL).	124
6.24.	Vereinfachte, nicht-maßstäbliche Darstellung des realen Fernwärmenetzes 4 mit Druckerhöhungs-, Speicher- und Wärmeerzeugungsanlagen.	124
7.1.	In <i>SIR 3S</i> [®] simulierte, normierte Gesamtleistung in Fernwärmenetz 1 im Vergleich zur Vorhersage an Beispieltag 1 im Januar.	132
7.2.	Vereinfachte Darstellung der Wärmeerzeugungsstation in Fernwärmenetz 1.	132
7.3.	Wärmeerzeugungsstation nach der ersten vereinfachten <i>SIR 3S</i> [®] Simulation.	133

7.4. Ent- und Beladen des Heißwasserspeichers in Fernwärmenetz 1.	134
7.5. Ent- und Beladen des Wärmespeichers in Fernwärmenetz 1.	135
7.6. Normierte Ergebnisse der Leistungsoptimierung in Fernwärmenetz 1.	136
7.7. Normierte Ergebnisse einer Validierungsrechnung mit erweiterter <i>SIR 3S</i> [®] Stationsmodellierung.	136
7.8. Normierte Vorlauftemperaturen und Wärmeverluste an Beispieltag 1 im Januar vor (IST) und nach (OPT) der Vorlauftempero-optimierung.	138
7.9. Simulationsergebnisse für Beispieltag 1 im Januar vor (IST) und nach (OPT) der stationären Vorlauftempero-optimierung von Fernwärmenetz 1.	139
7.10. Simulationsergebnisse für Beispieltag 1 im Januar vor (IST) und nach (OPT) einer Optimierung der Wärmeeinspeisung und Pumpenleistung.	140
7.11. Normierte Simulationsergebnisse für Beispieltag 1 im Januar vor (IST) und nach (OPT) der instationären Vorlauftempero-optimierung.	142
7.12. Normierte Simulationsergebnisse mit wirtschaftlich optimalen Sollwerten für Fernwärmenetz 2 an Beispieltag 1 im Februar.	145
7.13. Schematische Darstellung der Optimierung von Fernwärmenetz 2.	146
7.14. Normierte Simulationsergebnisse für Beispieltag 1 im Februar nach stationärer Optimierung von Fernwärmenetz 2.	147
7.15. Normierte stationäre und instationäre Simulationsergebnisse nach Optimierung von Fernwärmenetz 2 an Beispieltag 1 im Februar.	149
7.16. Normierte gesamte Einspeiseleistung in Fernwärmenetz 2 an Beispieltag 1 (Februar).	150
7.17. Normierte Simulationsergebnisse nach Optimierung von Fernwärmenetz 2 an Beispieltag 1 im Februar mit Leistungsvorgabe aus Prozessdaten.	152
7.18. Normierte Leistungen nach Optimierung von Fernwärmenetz 2 an Beispieltag 1 im Februar mit Leistungsvorgabe aus Prozessdaten und An-Vorgabe für das HW.	153
7.19. Normierte Simulationsergebnisse nach Minimierung der Wärmeverluste von Fernwärmenetz 2 an Beispieltag 1 im Februar.	155
7.20. Betriebsarten der DEA in Fernwärmenetz 3 an Beispieltag 1 (Februar).	158
7.21. Normierte Massenströme der DEA an Beispieltag 1 mit konstanter Betriebsart und mit konstanter Betriebsart sowie abgesenkter Leistung des HW 3 in Teilgebiet 2.	158
7.22. Normierte Simulationsergebnisse für Beispieltag 1 im Februar nach stationärer Optimierung von Fernwärmenetz 3.	160
7.23. Normierte Vorlauftemperaturen nach stationärer Optimierung von Fernwärmenetz 3 an Beispieltag 1 im Februar.	161
7.24. Normierte stationäre und instationäre Simulationsergebnisse nach Optimierung von Fernwärmenetz 3 an Beispieltag 1 im Februar.	163
7.25. Normierte Simulationsergebnisse nach Optimierung von Fernwärmenetz 3 an Beispieltag 1 im Februar mit Leistungsvorgabe aus Prozessdaten.	165
A.1. Lernkurven für die neuronalen Netze (NN) von Fernwärmenetz 4.	207
A.2. Normierte Simulationsergebnisse für Beispieltag 1 im Februar nach stationärer Optimierung von Fernwärmenetz 4.	214
A.3. Normierte Vorlauftemperaturen nach stationärer Optimierung von Fernwärmenetz 4 an Beispieltag 1 im Februar.	215
A.4. Normierte Simulationsergebnisse nach stationärer Optimierung von Fernwärmenetz 4 an Beispieltag 2 im Mai mit Leistungsvorgabe aus Prozessdaten.	216

E. Tabellenverzeichnis

6.1. Auswertung verschiedener Machine Learning Modelle mit Testdaten für Netz 1.	88
6.2. Abweichung verschiedener Machine Learning Modelle zu $SIR\ 3S^{\circledast}$ an Tag 1.	89
6.3. Abweichung verschiedener Machine Learning Modelle zu $SIR\ 3S^{\circledast}$ an Tag 3.	89
6.4. Bewertung neuronaler Netze mit Testdaten für Fernwärmenetz 1.	91
6.5. Abweichung verschiedener neuronaler Netze zu $SIR\ 3S^{\circledast}$ an Tag 1.	92
6.6. Abweichung verschiedener neuronaler Netze zu $SIR\ 3S^{\circledast}$ an Tag 3.	93
6.7. Bewertung des neuronalen Netzes mit 110, 88 und 44 Neuronen basierend auf Testdaten mit verschiedenen Aktivierungsfunktionen für Fernwärmenetz 1.	94
6.8. Abweichung des neuronalen Netzes mit 110, 88 und 44 Neuronen sowie verschiedenen Aktivierungsfunktionen zu $SIR\ 3S^{\circledast}$ an Tag 1 im Januar und 3 im Juli.	94
6.9. Abweichung des neuronalen Netzes mit 110, 88 und 44 Neuronen mit unterschiedlicher Datenskalierung zu $SIR\ 3S^{\circledast}$ an Tag 1 im Januar und 3 im Juli.	94
6.10. Mittlere 24h-Abweichungen der Nachberechnung von Fernwärmenetz 2 in $SIR\ 3S^{\circledast}$ an den drei Beispieltagen.	104
6.11. Auswertung verschiedener Machine Learning Modelle mit Testdaten von Netz 2, ohne Rücklauftemperatur als Eingang.	105
6.12. Auswertung verschiedener Machine Learning Modelle mit Testdaten von Netz 2, mit Rücklauftemperatur als Eingang.	106
6.13. Bewertung neuronaler Netze mit Testdaten für Fernwärmenetz 2 mit Rücklauftemperatur als Eingang nach 1000 Epochen Training.	107
6.14. Abweichung verschiedener neuronaler Netze für Fernwärmenetz 2 zu $SIR\ 3S^{\circledast}$ für zehn Simulationstage basierend auf Tag 1 im Februar.	108
6.15. Abweichung verschiedener neuronaler Netze für Fernwärmenetz 2 zu $SIR\ 3S^{\circledast}$ für zehn Simulationstage basierend auf Tag 3 im Juli.	108
6.16. Durchschnittliche Abweichung der neuronalen Netze zu $SIR\ 3S^{\circledast}$ für Fernwärmenetz 2	109
6.17. Maximale absolute und relative Abweichungen der neuronalen Netze zu $SIR\ 3S^{\circledast}$ für das aggregierte Netz 2 und zehn Simulationstage basierend auf Beispieltag 1 im Februar.	110
6.18. Maximale absolute und relative Abweichungen der neuronalen Netze zu $SIR\ 3S^{\circledast}$ für das aggregierte Netz 2 für zehn Simulationstage basierend auf Beispieltag 3 im Juli. .	111
6.19. Abweichungen des neuronalen Netzes zu den Testdaten der HKW-Rücklauftemperatur-Vorhersage in Fernwärmenetz 2.	113
6.20. Mittlere 24h-Abweichungen der Nachberechnung von Fernwärmenetz 3 in $SIR\ 3S^{\circledast}$ an den drei Beispieltagen.	116
6.21. Bewertung neuronaler Netze mit Rücklauftemperatur mit Testdaten für Fernwärmenetz 3 nach 1000 Epochen Training.	118
6.22. Abweichung verschiedener neuronaler Netze für Fernwärmenetz 3 zu $SIR\ 3S^{\circledast}$ für zehn Simulationstage basierend auf Tag 1 im Februar.	119
6.23. Abweichung verschiedener neuronaler Netze für Fernwärmenetz 3 zu $SIR\ 3S^{\circledast}$ für zehn Simulationstage basierend auf Tag 3 im Juli.	119
6.24. Optimierungsdauer der neuronalen Netze für Fernwärmenetz 3 an den drei Beispieltagen.	119
6.25. Durchschnittliche Abweichung der neuronalen Netze zu $SIR\ 3S^{\circledast}$ für Fernwärmenetz 3.	120
6.26. Maximale absolute und relative Abweichungen des neuronalen Netzes mit Rücklauftemperatur zu $SIR\ 3S^{\circledast}$ für Netz 3 und zehn Simulationstage basierend auf Tag 1.	121

6.27. Mittlere 24h-Abweichungen der Nachberechnung von Fernwärmenetz 4 in $SIR\ 3S^{\circledast}$ an den drei Beispieltagen.	125
6.28. Durchschnittliche Abweichung der neuronalen Netze zu $SIR\ 3S^{\circledast}$ für Fernwärmenetz 4	125
7.1. Optimierungsdauer für Netz 2 an den drei Beispieltagen mit Leistungsvorgabe aus der Vortags-Prognose.	148
7.2. Optimierungsdauer für Netz 2 an den drei Beispieltagen mit Leistungsvorgabe aus Prozessdaten.	151
7.3. Optimierungsdauer für Netz 3 an den drei Beispieltagen mit Leistungsvorgabe aus der Vortags-Prognose.	162
7.4. Optimierungsdauer für Netz 3 an den drei Beispieltagen mit Leistungsvorgabe aus Prozessdaten.	164
A.1. Bewertung neuronaler Netze mit Testdaten für Fernwärmenetz 1 nach 100 Epochen Training.	175
A.2. Bewertungsscore neuronaler Netze für Fernwärmenetz 1 in Tabelle A.1.	177
A.3. Bewertung neuronaler Netze mit Testdaten für Fernwärmenetz 1 nach 1000 Epochen Training.	177
A.4. Bewertungsscore neuronaler Netze für Fernwärmenetz 1 in Tabelle A.3.	178
A.5. Maximale absolute Abweichungen des neuronalen Netzes zu $SIR\ 3S^{\circledast}$ für das aggregierte Netz 1.	179
A.6. Maximale relative Abweichungen des neuronalen Netzes zu $SIR\ 3S^{\circledast}$ für das aggregierte Netz 1.	179
A.7. Maximale absolute Abweichungen des neuronalen Netzes zu $SIR\ 3S^{\circledast}$ für das vollständige Netz 1.	180
A.8. Maximale relative Abweichungen des neuronalen Netzes zu $SIR\ 3S^{\circledast}$ für das vollständige Netz 1.	180
A.9. Bewertung neuronaler Netze mit Testdaten für Fernwärmenetz 2 nach 100 Epochen Training.	181
A.10. Bewertungsscore neuronaler Netze für Fernwärmenetz 2.	185
A.11. Ausgangsgrößen der neuronalen Netze für Fernwärmenetz 2 und Verwendung für die Berechnung der relativen Modellgenauigkeit.	186
A.12. Maximale absolute und relative Abweichungen der neuronalen Netze zu $SIR\ 3S^{\circledast}$ für das aggregierte Netz 2 und für zehn Simulationstage basierend auf Beispieltag 1 (Februar).	187
A.13. Maximale absolute und relative Abweichungen der neuronalen Netze zu $SIR\ 3S^{\circledast}$ für das aggregierte Netz 2 und für zehn Simulationstage basierend auf Beispieltag 2 (Mai).	188
A.14. Maximale absolute und relative Abweichungen der neuronalen Netze zu $SIR\ 3S^{\circledast}$ für das aggregierte Netz 2 und für zehn Simulationstage basierend auf Beispieltag 3 (Juli).	189
A.15. Bewertung neuronaler Netze mit Testdaten für Fernwärmenetz 3 nach 100 Epochen Training.	190
A.16. Bewertungsscore neuronaler Netze für Fernwärmenetz 3.	195
A.17. Ausgangsgrößen der neuronalen Netze für Fernwärmenetz 3 und Verwendung für die Berechnung der relativen Modellgenauigkeit.	196
A.18. Maximale absolute und relative Abweichungen der neuronalen Netze zu $SIR\ 3S^{\circledast}$ für Netz 3 und für zehn Simulationstage basierend auf Beispieltag 1 im Februar.	197
A.19. Maximale absolute und relative Abweichungen der neuronalen Netze zu $SIR\ 3S^{\circledast}$ für Netz 3 und für zehn Simulationstage basierend auf Beispieltag 2 im Mai.	199
A.20. Maximale absolute und relative Abweichungen der neuronalen Netze zu $SIR\ 3S^{\circledast}$ für Netz 3 und für zehn Simulationstage basierend auf Beispieltag 3 im Juli.	200
A.21. Mittlere 24h-Abweichungen der Nachberechnung von Fernwärmenetz 4 in $SIR\ 3S^{\circledast}$ an den drei Beispieltagen.	202

A.22.Ausgangsgrößen der neuronalen Netze für Fernwärmenetz 3 und Verwendung für die Berechnung der relativen Modellgenauigkeit.	204
A.23.Bewertung neuronaler Netze mit Rücklaufftemperatureingang mit Testdaten für Fernwärmenetz 4 nach 1000 Epochen Training.	206
A.24.Abweichung verschiedener neuronaler Netze für Fernwärmenetz 4 zu <i>SIR 3S</i> [®] für zehn Simulationstage basierend auf Beispieltag 1 im Februar.	206
A.25.Abweichung verschiedener neuronaler Netze für Fernwärmenetz 4 zu <i>SIR 3S</i> [®] für zehn Simulationstage basierend auf Beispieltag 2 im Mai.	206
A.26.Durchschnittliche Abweichung der neuronalen Netze zu <i>SIR 3S</i> [®] für Fernwärmenetz 4	207
A.27.Maximale absolute und relative Abweichungen der neuronalen Netze zu <i>SIR 3S</i> [®] für Netz 4 und zehn Simulationstage basierend auf Beispieltag 1 im Februar.	208
A.28.Maximale absolute und relative Abweichungen der neuronalen Netze zu <i>SIR 3S</i> [®] für Netz 4 und zehn Simulationstage basierend auf Beispieltag 2 im Mai.	210
A.29.Optimierungsdauer für Netz 4 an den zwei Beispieltagen mit Leistungsvorgabe aus der Vortags-Prognose.	215
A.30.Optimierungsdauer für Netz 4 an den zwei Beispieltagen mit Leistungsvorgabe aus Prozessdaten.	216

F. Literaturverzeichnis

- [1] *Gesetz zur Reduzierung und zur Beendigung der Kohleverstromung und zur Änderung weiterer Gesetze (Kohleausstiegsgesetz)*, BGBl. I S. 1818.. – http://www.bgbl.de/xaver/bgbl/start.xav?startbk=Bundesanzeiger_BGBl&jumpTo=bgbl1120s1818.pdf – Zugriff: 17.10.2022
- [2] ABADI, M. ; AGARWAL, A. ; BARHAM, P. ; BREVDO, E. ; CHEN, Z. ; CITRO, C. ; CORRADO, G. S. ; DAVIS, A. ; DEAN, J. ; DEVIN, M. ; GHEMAWAT, S. ; GOODFELLOW, I. ; HARP, A. ; IRVING, G. ; ISARD, M. ; JIA, Y. ; JOZEFOWICZ, R. ; KAISER, L. ; KUDLUR, M. ; LEVENBERG, J. ; MANE, D. ; MONGA, R. ; MOORE, S. ; MURRAY, D. ; OLAH, C. ; SCHUSTER, M. ; SHLENS, J. ; STEINER, B. ; SUTSKEVER, I. ; TALWAR, K. ; TUCKER, P. ; VANHOUCHE, V. ; VASUDEVAN, V. ; VIEGAS, F. ; VINYALS, O. ; WARDEN, P. ; WATTENBERG, M. ; WICKE, M. ; YU, Y. ; ZHENG, X. : *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*. 2016. – <https://doi.org/10.48550/arXiv.1603.04467>
- [3] ABDOLLAHI, E. ; WANG, H. ; RINNE, S. ; LAHDELMA, R. : *Optimization of energy production of a CHP plant with heat storage*. 2014 IEEE Green Energy and Systems Conference (IGESC), Long Beach, CA (USA), November 2014. – <https://doi.org/10.1109/IGESC.2014.7018636>
- [4] ABOKERSH, M. H. ; VALLÈS, M. ; CABEZA, L. F. ; BOER, D. : *A framework for the optimal integration of solar assisted district heating in different urban sized communities: A robust machine learning approach incorporating global sensitivity analysis*. Applied Energy, Nr. 267, 114903, Juni 2020. – <https://doi.org/10.1016/j.apenergy.2020.114903>
- [5] ABOKERSH, M. H. ; VALLÈS, M. ; JIMÉNEZ, L. ; BOER, D. : *Cost-Effective Processes of Solar District Heating System Based on Optimal Artificial Neural Network*. Computer Aided Chemical Engineering, Nr. 48, S. 403-408, Oktober 2020. – <https://doi.org/10.1016/B978-0-12-823377-1.50068-9>
- [6] ABUGABBARA, M. ; JAVED, S. ; BAGGE, H. ; JOHANSSON, D. : *Bibliographic analysis of the recent advancements in modeling and co-simulating the fifth-generation district heating and cooling systems*. Energy and Buildings, Nr. 224, Oktober 2020. – <https://doi.org/10.1016/j.enbuild.2020.110260>
- [7] ALLEGRINI, J. ; OREHOUNIG, K. ; MAVROMATIDIS, G. ; RUESCH, F. ; DORER, V. ; EVINS, R. : *A review of modelling approaches and tools for the simulation of district-scale energy systems*. Renewable and Sustainable Energy Reviews, Nr. 52, S. 1391-1404, Dezember 2015. – <https://doi.org/10.1016/j.rser.2015.07.123>
- [8] ANDRESEN, L. ; DUBUCQ, P. ; PENICHE GARCIA, R. ; ACKERMANN, G. ; KATHER, A. ; SCHMITZ, G. : *Status of the TransiEnt Library : Transient simulation of coupled energy networks with high share of renewable energy*. Proceedings of the 11th International Modelica Conference, Versailles (Frankreich), September 2015. – <http://doi.org/10.3384/ecp15118695>
- [9] BACH, B. ; WERLING, J. ; OMMEN, T. ; MÜNSTER, M. ; MORALES, J. M. ; ELMEGAARD, B. : *Integration of large-scale heat pumps in the district heating systems of Greater Copenhagen*. Energy, Nr. 107, S. 321-334, Juli 2016. – <https://doi.org/10.1016/j.energy.2016.04.029>

- [10] BASCIOTTI, D. ; JUDEX, F. ; POL, O. ; SCHMIDT, R.-R. : *Sensible heat storage in district heating networks: a novel control strategy using the network as storage*. 6th International Renewable Energy Storage Conference and Exhibition (IRES 2011), Berlin (Deutschland), November 2011
- [11] BDEW BUNDESVERBAND DER ENERGIE- UND WASSERWIRTSCHAFT E. V. : *Entwicklung des Wärmeverbrauchs in Deutschland. Basisdaten und Einflussfaktoren*. 6. Ausgabe, 16.03.2022. – https://www.bdew.de/media/documents/20220511_W%C3%A4rmeverbrauchsanalyse_Foliensatz_2022_final.pdf – Zugriff: 14.10.2022
- [12] BDEW BUNDESVERBAND DER ENERGIE- UND WASSERWIRTSCHAFT E. V. ; VERBAND KOMMUNALER UNTERNEHMEN E. V. (VKU) ; GEODE - GROUPEMENT EUROPÉEN DES ENTREPRISES ET ORGANISMES DE DISTRIBUTION D'ÉNERGIE, E. : *BDEW/VKU/GEODE - Leitfaden. Abwicklung von Standardlastprofilen Gas*. 31.03.2020. – https://www.bdew.de/media/documents/20200331_KoV_XI_LF_SLP_Gas_clean_final.pdf – Zugriff: 14.10.2022
- [13] BENONYSSON, A. ; BØHM, B. ; RAVN, H. F. : *Operational Optimization in a District Heating System*. Energy Conversion and Management, Nr. 36 (5), S. 297-314, Mai 1995. – [https://doi.org/10.1016/0196-8904\(95\)98895-T](https://doi.org/10.1016/0196-8904(95)98895-T)
- [14] BERGSTEINSSON, H. G. ; SKOV NIELSEN, T. ; KLOPPENBORG MØLLER, J. ; BEN AMER, S. ; DOMINIKOVIĆ, D. F. ; MADSEN, H. : *Use of smart meters as feedback for district heating temperature control*. Energy Reports, Nr. 7 (4), S. 213-221s, September 2021. – <https://doi.org/10.1016/j.egyr.2021.08.153>
- [15] BERTHOLD, J. ; JULIN, A. : *The Effects of Lowered Delivery Temperatures in District Heating - A Quantitative Study*. Masterthesis, KTH Royal Institute of Technology, School of Industrial Engineering and Management, September 2015. – <http://kth.diva-portal.org/smash/record.jsf?pid=diva2%3A876224&dsid=-6839> – Zugriff: 14.10.2022
- [16] BISCANI, F. ; IZZO, D. : *A parallel global multiobjective framework for optimization: pagmo*. Journal of Open Source Software, Nr. 5 (53), 2338, September 2022. – <https://doi.org/10.21105/joss.02338>
- [17] BÖSWIRTH, L. ; BSCHORER, S. : *Technische Strömungslehre: Lehr- und Übungsbuch*. 10. Auflage. Springer Vieweg, 2014. – ISBN 978-3-658-05667-4
- [18] BOTT, A. ; FRIEDRICH, P. ; REHLICH, L. ; STEINKE, F. : *Model Reduction for Heat Grid State Estimation*. 2021 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe), Espoo (Finnland), Oktober 2021. – <https://doi.org/10.1109/ISGTEurope52324.2021.9640025>
- [19] BOUKOUVALA, F. ; MISENER, R. ; FLOUDAS, C. A. : *Global optimization advances in Mixed-Integer Nonlinear Programming, MINLP, and Constrained Derivative-Free Optimization, CDFO*. European Journal of Operational Research, Nr. 252 (3), S. 701-727, August 2016. – <https://doi.org/10.1016/j.ejor.2015.12.018>
- [20] BUFFA, S. ; SOPPELSA, A. ; PIPICIELLO, M. ; HENZE, G. ; FEDRIZZI, R. : *Fifth-generation district heating and cooling substations: Demand response with artificial neural network-based model predictive control*. Energies, Nr. 13 (17), 4339, August 2020. – <https://doi.org/10.3390/en13174339>
- [21] BUORO, D. ; PINAMONTI, P. ; REINI, M. : *Optimization of a Distributed Cogeneration System with solar district heating*. Applied Energy, Nr. 124, S. 298-308, Juli 2014. – <https://doi.org/10.1016/j.apenergy.2014.02.062>

- [22] BÜSKENS, C. ; WASSEL, D. : *The ESA NLP Solver WORHP*. In: Fasano, G., Pintér, J. (Eds.), *Modeling and Optimization in Space Engineering*. Springer Optimization and Its Applications, Nr. 73, Springer, New York, NY (USA), 2012. – https://doi.org/10.1007/978-1-4614-4469-5_4
- [23] CADAU, N. ; DE LORENZI, A. ; GAMBAROTTA, A. ; MORINI, M. ; SALETTI, C. : *A Model-in-the-Loop application of a Predictive Controller to a District Heating system*. Energy Procedia, Nr. 148, S. 352-359, August 2018. – <https://doi.org/10.1016/j.egypro.2018.08.088>
- [24] CARPANETO, E. ; LAZZERONI, P. ; REPETTO, M. : *Optimal integration of solar energy in a district heating network*. Renewable Energy, Nr. 75, S. 714-721, März 2015. – <https://doi.org/10.1016/j.renene.2014.10.055>
- [25] CERBE, A. : *Verbundprojekt Bessere Ausnutzung von Fernwärmeanlagen. Pilotprojekt HEW Hamburg: Prozessnahe Einsatzoptimierung mit BoFiT unter Berücksichtigung der Netzrestriktionen*. Hamburgische Electricitäts-Werke AG, Hamburg (Deutschland), 2002. – <http://edok01.tib.uni-hannover.de/edoks/e01fb02/357623436.pdf> – Zugriff: 14.10.2022
- [26] CHEN, T. ; GUESTRIN, C. : *XGBoost: A Scalable Tree Boosting System*. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, California (USA), S. 785 - 794, August 2016. – <https://doi.org/10.1145/2939672.2939785>
- [27] COCCIA, G. ; MUGNINI, A. ; POLONARA, F. ; ARTECONI, A. : *Artificial-neural-network-based model predictive control to exploit energy flexibility in multi-energy systems comprising district cooling*. Energy, Nr. 222, 119958, Mai 2021. – <https://doi.org/10.1016/j.energy.2021.119958>
- [28] CONEJO, A. J. ; BARINGO, L. : *Unit Commitment and Economic Dispatch*. In: *Power System Operations*. 1. Auflage. Springer Vieweg, Cham (Schweiz), 2018. – ISBN 978-3-319-69407-8
- [29] CORREA-JULLIAN, C. ; LÓPEZ DROGUETT, E. ; CARDEMIL, J. M.: *Operation scheduling in a solar thermal system: A reinforcement learning-based framework*. Applied Energy, Nr. 268, 114943, Juni 2020. – <https://doi.org/10.1016/j.apenergy.2020.114943>
- [30] COX, S. J. ; KIM, D. ; CHO, H. ; MAGO, P. : *Real time optimal control of district cooling system with thermal energy storage using neural networks*. Applied Energy, Nr. 238, S. 466-480, März 2019. – <https://doi.org/10.1016/j.apenergy.2019.01.093>
- [31] DEL HOYO ARCE, I. ; LÓPEZ, S. H. ; PEREZ, S. L. ; RÄMÄ, M. ; KLOBUT, K. ; FEBRES, J. A.: *Models for fast modelling of district heating and cooling networks*. Renewable and Sustainable Energy Reviews, Nr. 82 (2), S. 1863-1873, Februar 2018. – <https://doi.org/10.1016/j.rser.2017.06.109>
- [32] DITTMAR, R. ; PFEIFFER, B.-M. : *Modellbasierte prädiktive Regelung: Eine Einführung für Ingenieure*. 1. Auflage. Oldenbourg Wissenschaftsverlag GmbH, 2004. – ISBN 978-3-486-27523-2
- [33] DUBUCQ, P. ; ACKERMANN, Prof. Dr.-Ing. G. : *Frequency Control in Coupled Energy Systems with High Penetration of Renewable Energies*. 2015 International Conference on Clean Electrical Power (ICCEP), Taormina (Italien), Juni 2015. – <https://doi.org/10.1109/ICCEP.2015.7177643>
- [34] DUBUCQ, P. ; ACKERMANN, Prof. Dr.-Ing. G. : *Optimal Use of Energy Storage Potentials in a Renewable Energy System with District Heating*. Energy Procedia, Nr. 135, S. 158-171, Oktober 2017. – <https://doi.org/10.1016/j.egypro.2017.09.499>

- [35] ENDRES, S. C. ; SANDROCK, C. ; FOCKE, W. W.: *A simplicial homology algorithm for Lipschitz optimisation*. Journal of Global Optimization, Nr. 72, S. 181-217, März 2018. – <https://doi.org/10.1007/s10898-018-0645-y>
- [36] ERTEL, W. : *Grundkurs Künstliche Intelligenz: Eine praxisorientierte Einführung*. In: W. Bibel, R. Kruse, B. Nebel (Hrsg.), Computational Intelligence. 4., überarbeitete Auflage. Springer Fachmedien Wiesbaden, 2015. – ISBN 978-3-658-13549-2
- [37] FANG, T. ; LAHDELMA, R. : *Optimization of combined heat and power production with heat storage based on sliding time window method*. Applied Energy, Nr. 162, S. 723-732, Januar 2016. – <https://doi.org/10.1016/j.apenergy.2015.10.135>
- [38] FANTI, M. P. ; MANGINI, A. M. ; ROCCOTELLI, M. ; UKOVICH, W. : *A District Energy Management Based on Thermal Comfort Satisfaction and Real-Time Power Balancing*. IEEE Transactions on Automation Science and Engineering, Nr. 12 (4), S. 1271-1284, Oktober 2015. – <https://doi.org/10.1109/TASE.2015.2472956>
- [39] FEINDT, O. ; KÄSER, G. ; FISCHER, M. ; DIRNER, S. ; NOLD, K. ; SACHSENHAUSER, M. ; MICHELS, K. : *GEOMATIC - Optimale Verteilung der Wärmeeinspeisung in Fernwärmenetzen*. Euro Heat & Power, Heft 4-5, 2022. ISSN 0949-166X-D9790F
- [40] FEINDT, O. ; SACHSENHAUSER, M. ; NOLD, K. ; DIRNER, S. ; KÄSER, G. ; FISCHER, M. ; MICHELS, K. : *Modellprädiktive Regelung für die optimale Verteilung der Wärmeeinspeisung in Fernwärmenetzen*. 54. Kraftwerkstechnisches Kolloquium, Dresden. In: M. Beckmann, A. Hurtado (Hrsg.), Kraftwerkstechnik 2022 - Power Plant Technology. Innovation- und Kreislaufwirtschaft Sachsen e. V., Freiberg, 2022. – ISBN 978-3-00-072104-5
- [41] FREDERIKSEN, S. ; WERNER, S. : *District Heating and Cooling*. 1. Auflage. Studentlitteratur AB, 2013. – ISBN 978-91-44-08530-2
- [42] FROCHTE, J. : *Maschinelles Lernen: Grundlagen und Algorithmen in Python*. 3. Auflage. Carl Hanser Verlag, 2021. – ISBN 978-3-446-46355-4
- [43] FROTSCHER, O. ; OPPELT, T. ; URBANECK, T. ; OTTO, S. ; HEINRICH, I. ; SCHMIDT, A. ; GÖSCHEL, T. ; UHLIG, U. : *Heating Network Navigator - Software Solution for Optimal Operation of Complex DH systems*. Euro Heat & Power (Englische Ausgabe), Nr. 1 (2020), Januar 2020
- [44] FROTSCHER, O. ; OPPELT, T. ; URBANECK, T. ; OTTO, S. ; HEINRICH, I. ; SCHMIDT, A. ; GÖSCHEL, T. ; UHLIG, U. ; FREY, H. : *Software-in-the-loop-simulation of a district heating system as test environment for a sophisticated operating software*. Proceedings of the 9th International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH 2019), Prag (Tschechien), Juli 2019. – <https://doi.org/10.5220/0007809602230230>
- [45] GAMBINO, G. ; VERRILLI, F. ; MEOLA, D. ; HIMANKA, M. ; PALMIERI, G. ; DEL VECCHIO, C. ; GLIELMO, L. : *Model predictive control for optimization of combined heat and electric power microgrid*. IFAC Proceedings Volumes, Nr. 47 (3), S. 2201-2206, August 2014. – <https://doi.org/10.3182/20140824-6-ZA-1003.02354>
- [46] GAMBINO, G. ; VERRILLI, F. ; CANELLI, M. ; RUSSO, A. ; HIMANKA, M. ; SASSO, M. ; SRINIVASAN, S. ; DEL VECCHIO, C. ; GLIELMO, L. : *Optimal operation of a District Heating power plant with Thermal Energy Storage*. 2016 American Control Conference (ACC), Boston, MA (USA), August 2016. – <https://doi.org/10.1109/ACC.2016.7525266>

- [47] GIRAUD, L. ; BAVIÈRE, R. ; PAULUS, C. ; VALLÉE, M. ; ROBIN, J.-F. : *Dynamic Modelling, Experimental Validation and Simulation of a Virtual District Heating Network*. The 28th International Conference on Efficiency, Cost, Optimization, Simulation and Environmental Impact of Energy Systems, Pau (Frankreich), Juli 2015
- [48] GIRAUD, L. ; MERABET, M. ; BAVIERE, R. ; VALLÉE, M. : *Optimal Control of District Heating Systems using Dynamic Simulation and Mixed Integer Linear Programming*. Proceedings of the 12th International Modelica Conference, Prag (Tschechien), Mai 2017. – <http://dx.doi.org/10.3384/ecp17132141>
- [49] GLOROT, X. ; BENGIO, Y. : *Understanding the difficulty of training deep feedforward neural networks*. Proceedings of the 13th International Conference on Artificial Intelligence and Statistics, Sardinien (Italien), S. 249-256, 2010. – <https://proceedings.mlr.press/v9/glorot10a.html>
- [50] GOLLA, A. ; GEIS, J. ; LOY, T. ; STAUDT, P. ; WEINHARDT, C. : *An operational strategy for district heating networks: application of data-driven heat load forecasts*. Energy Informatics, Nr. 3 (S1), 22, Oktober 2020. – <https://doi.org/10.1186/s42162-020-00125-5>
- [51] GROSSWINDHAGER, S. ; VOIGT, A. ; KOZEK, M. : *Efficient Physical Modelling of District Heating Networks*. Proceedings of the IASTED International Conference on Modelling, Simulation and Optimization, Anaheim, CA (USA), August 2011. – <http://dx.doi.org/10.2316/P.2011.735-094>
- [52] GROSSWINDHAGER, S. ; VOIGT, A. ; KOZEK, M. : *Predictive Control of District Heating Network using Fuzzy DMC*. 2012 Proceedings of International Conference on Modelling, Identification and Control, Wuhan (China), S. 241-246, Juni 2012
- [53] GU, W. ; WU, Z. ; YUAN, X. : *Microgrid Economic Optimal Operation of the Combined Heat and Power System with Renewable Energy*. IEEE PES General Meeting, Minneapolis, MN (USA), Juli 2010. – <https://doi.org/10.1109/PES.2010.5590140>
- [54] GUELPA, E. ; TORO, C. ; SCIACOVELLI, A. ; MELLI, R. ; SCIUBBA, E. ; VERDA, V. : *Optimal operation of large district heating networks through fast fluid-dynamic simulation*. Energy, Nr. 102, S. 586-595, Mai 2016. – <https://doi.org/10.1016/j.energy.2016.02.058>
- [55] GUELPA, E. ; VERDA, V. : *Optimization of the thermal load profile in district heating networks through “virtual storage” at building level*. Energy Procedia, Nr. 101, S. 798-805, November 2016. – <https://doi.org/10.1016/j.egypro.2016.11.101>
- [56] HÄGG, R. : *Dynamic Simulation of District Heating Networks in Dymola*. Masterthesis, Lund University, Faculty of Engineering, 2016. – <https://lup.lub.lu.se/student-papers/search/publication/8883842> – Zugriff: 14.10.2022
- [57] HE, H. : *The State of Machine Learning Frameworks in 2019*. The Gradient, Oktober 2019. – <https://thegradient.pub/state-of-ml-frameworks-2019-pytorch-dominates-research-tensorflow-dominates-industry/> – Zugriff: 14.10.2022
- [58] HE, K. ; ZHANG, X. ; REN, S. ; SUN, J. : *Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification*. 2015 IEEE International Conference on Computer Vision (ICCV), Santiago (Chile), S. 1026-1034, Dezember 2015. – <https://doi.org/10.1109/ICCV.2015.123>
- [59] HEIJDE, B. van d. ; VANDERMEULEN, A. ; SALENBIEN, R. ; HELSEN, L. : *Integrated Optimal Design and Control of Fourth Generation District Heating Networks with Thermal Energy Storage*. Energies, Nr. 12 (14), 2766, Juli 2019. – <https://doi.org/10.3390/en12142766>

- [60] HOLMGREN, MAGNUS: *X Steam, Thermodynamic properties of water and steam.* – <https://de.mathworks.com/matlabcentral/fileexchange/9817-x-steam-thermodynamic-properties-of-water-and-steam> – Zugriff: 20.02.2023 – [Software]
- [61] ICKING, M. : *Zur Modellierung des dynamischen Betriebs von Fernwärmesystemen.* 1. Auflage. Hartung-Gorre Verlag Konstanz, 1995. – ISBN 978-3-89191-952-1
- [62] IDOWU, S. ; AHLUND, C. ; SCHELEN, O. : *Machine learning in district heating system energy optimization.* 2014 IEEE International Conference on Pervasive Computing and Communication Workshops (PERCOM WORKSHOPS), Budapest (Ungarn), S. 224-227, Mai 2014. – <https://doi.org/10.1109/PerComW.2014.6815206>
- [63] JIE, P. ; ZHU, N. ; LI, D. : *Operation optimization of existing district heating systems.* Applied Thermal Engineering, Nr. 78, S. 278-288, März 2015. – <https://doi.org/10.1016/j.applthermaleng.2014.12.070>
- [64] JOHANSSON, C. ; VANHOUDT, D. ; BRAGE, J. ; GEYSEN, D. : *Real-time grid optimisation through digitalisation - Results of the STORM project.* Energy Procedia, Nr. 149, S. 246-255, September 2018. – <https://doi.org/10.1016/j.egypro.2018.08.189>
- [65] JUNGWIRTH, J. : *Lastmanagement in Gebäuden: Entwicklung einer modellprädiktiven Regelung mit einem adaptiven Gebäudemodell zur Flexibilisierung der Wärme- und Kälteversorgung von Gebäuden.* Dissertation, Technische Universität München, Fakultät für Elektrotechnik und Informationstechnik, 2015. – <https://mediatum.ub.tum.de/1221398> – Zugriff: 14.10.2022
- [66] KEÇEBAŞ, A. ; ALKAN, M. A. ; YABANOVA, I. ; YUMURTACI, M. : *Energetic and economic evaluations of geothermal district heating systems by using ANN.* Energy Policy, Nr. 56, S. 558-567, Mai 2013. – <https://doi.org/10.1016/j.enpol.2013.01.039>
- [67] KEÇEBAŞ, A. ; YABANOVA, I. : *Thermal monitoring and optimization of geothermal district heating systems using artificial neural network: A case study.* Energy and Buildings, Nr. 50, S. 339-346, July 2012. – <https://doi.org/10.1016/j.enbuild.2012.04.002>
- [68] KÖCHER, R. : *Beitrag zur Berechnung und Auslegung von Fernwärmenetzen.* Dissertation, Technische Universität Berlin, Fachbereich Verfahrenstechnik, Umwelttechnik, Werkstoffwissenschaften, 2000. – <https://depositonce.tu-berlin.de/items/ac3fe775-b07e-45f7-8a50-1851908f2a93> – Zugriff: 14.10.2022
- [69] KONSTANTIN, P. : *Praxisbuch der Fernwärmeversorgung.* 1. Auflage. Springer Vieweg Berlin, Heidelberg, 2018. – ISBN 978-3-662-55911-6
- [70] KOZEK, M. : *BioNetControl-Systems Regelungs- und Optimierungssystem für den energieeffizienten Betrieb von Fernwärmenetzen an Biomasseanlagen.* Blue Globe Report, Erneuerbare Energien #2/2012, 2012. – <https://www.klimafonds.gv.at/wp-content/uploads/sites/6/BGR0022012EEeneueEnergien2020.pdf> – Zugriff: 14.10.2022
- [71] KRAFT, D. : *A Software Package for Sequential Quadratic Programming.* Forschungsbericht DFVLR-FB 88-28, Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt, Institut für Dynamik der Flugsysteme, Oberpfaffenhofen (Deutschland), 1988
- [72] KRUSE, R. ; BORGELT, C. ; BRAUNE, C. ; KLAWONN, F. ; MOEWES, C. ; STEINBRECHER, M. : *Computational Intelligence: Eine methodische Einführung in Künstliche Neuronale Netze, Evolutionäre Algorithmen, Fuzzy-Systeme und Bayes-Netze.* In: W. Bibel, R. Kruse, B. Nebel (Hrsg.), Computational Intelligence. 2. Auflage. Springer Fachmedien Wiesbaden, 2015. – ISBN 978-3-658-10904-2

- [73] KUDELA, L. ; CHÝLEK, R. ; POSPÍŠIL, J. : *Efficient Integration of Machine Learning into District Heating Predictive Models*. Energies, Nr. 13 (23), 6381, Dezember 2020. – <https://doi.org/10.3390/en13236381>
- [74] LAAKKONEN, L. ; KORPELA, T. ; KAIVOSOJA, J. ; VILKKO, M. ; MAJANNE, Y. ; NURMORANTA, M. : *Predictive Supply Temperature Optimization of District Heating Networks Using Delay Distributions*. Energy Procedia, Nr. 116, S. 297-309, Juni 2017. – <https://doi.org/10.1016/j.egypro.2017.05.076>
- [75] LARSEN, H. V. ; BØHM, B. ; WIGBELS, M. : *A comparison of aggregated models for simulation and operational optimisation of district heating networks*. Energy Conversion and Management, Nr. 45 (7-8), S. 1119-1139, Mai 2004. – <https://doi.org/10.1016/j.enconman.2003.08.006>
- [76] LE-COZ, A. ; NABIL, T. ; COURTOT, F. : *Towards Optimal District Heating Temperature Control in China with Deep Reinforcement Learning*. 34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver (Kanada), Dezember 2020. – <https://doi.org/10.48550/arXiv.2012.09508>
- [77] LEE, D. ; YOON, S. M. ; LEE, J. ; KIM, K. ; SONG, S. H. : *Applying deep learning to the heat production planning problem in a district heating system*. Energies, Nr. 13 (24), 6641, Dezember 2020. – <https://doi.org/10.3390/en13246641>
- [78] LJUNG, L. : *System Identification: Theory for the User*. 2. Ausgabe. Prentice Hall PTR, 1999. – ISBN 978-0-136-56695-3
- [79] LOEWEN, A. : *Entwicklung eines Verfahrens zur Aggregation komplexer Fernwärmenetze*. Fraunhofer IRB Verlag, 2001. – <http://publica.fraunhofer.de/dokumente/N-6894.html>
- [80] LORENZEN, P. : *Das Wärmenetz als Speicher im Smart Grid: Betriebsführung eines Wärmenetzes in Kombination mit einem stromgeführten Heizkraftwerk*. Masterthesis, Hochschule für Angewandte Wissenschaften Hamburg, Fakultät Technik und Informatik, Mai 2014. – <https://reposit.haw-hamburg.de/handle/20.500.12738/6561> – Zugriff: 14.10.2022
- [81] LU, W. ; YANG, J. ; LI, Y. : *Control method based on fuzzy cognitive map and its application on district heating network*. 2010 International Conference on Intelligent Control and Information Processing, Dalian (China), S. 295-299, August 2010. – <https://doi.org/10.1109/ICICIP.2010.5564219>
- [82] LUNZE, Prof. Dr.-Ing. J. : *Regelungstechnik 1: Systemtheoretische Grundlagen, Analyse und Entwurf einschleifiger Regelungen*. 10., aktualisierte Auflage. Springer-Verlag Berlin Heidelberg, 2014. – ISBN 978-3-642-53909-1
- [83] MADSEN, H. ; SEJLING, K. ; SØGAARD, H. T. ; PALSSON, O. P. : *On flow and supply temperature control in district heating systems*. Heat Recovery Systems and CHP, Nr. 14 (6), S. 613-620, November 1994. – [https://doi.org/10.1016/0890-4332\(94\)90031-0](https://doi.org/10.1016/0890-4332(94)90031-0)
- [84] MALINOWSKI, P. ; ZIEMBICKI, P. : *Analysis of district heating network monitoring by neural networks classification*. Journal of Civil Engineering and Management, Nr. 12 (1), S. 21-28, 2006
- [85] MBIYDZENYUY, G. ; NOWACZYK, S. ; KNUTSSON, H. ; VANHOUDT, D. ; BRAGE, J. ; CALIKUS, E. : *Opportunities for machine learning in district heating*. Applied Sciences, Nr. 11 (13), 6112, Mai 2021. – <https://doi.org/10.3390/app11136112>
- [86] MERKERT, L. ; HAIME, A. A. ; HOHMANN, S. : *Optimal Scheduling of Combined Heat and Power Generation Units Using the Thermal Inertia of the Connected District Heating Grid as Energy Storage*. Energies, Nr. 12 (2), 266, Januar 2019. – <https://doi.org/10.3390/en12020266>

- [87] MOUSTAKIDIS, S. ; MEINTANIS, I. ; HALIKIAS, G. ; KARCANIAS, N. : *An Innovative Control Framework for District Heating Systems: Conceptualisation and Preliminary Results*. MDPI Resources, Nr. 8 (1), 27, Januar 2019. – <https://doi.org/10.3390/resources8010027>
- [88] MUGNINI, A. ; FERRACUTI, F. ; LORENZETTI, M. ; COMODI, G. ; ARTECONI, A. : *Advanced control techniques for CHP-DH systems: A critical comparison of Model Predictive Control and Reinforcement Learning*. Energy Conversion and Management, Nr. 15, 100264, August 2022. – <https://doi.org/10.1016/j.ecmx.2022.100264>
- [89] NAGY, A. ; KAZMI, H. ; CHEAIB, F. ; DRIESEN, J. : *Deep Reinforcement Learning for Optimal Control of Space Heating*. 4th Building Simulation and Optimization Conference (BSO 2018), Cambridge (UK), Mai 2018. – <https://doi.org/10.48550/arXiv.1805.03777>
- [90] NOCEDAL, J. ; WRIGHT, S. J.: *Numerical Optimization*. 2. Ausgabe. Springer New York, NY (USA), 2006. – ISBN 978-0-387-40065-1
- [91] NOLD, K. ; FEINDT, O. : *Vorstellung des F&E Verbundprojekts GEOmaRe - Teilprojekt Geomatic*. Der Geothermiekongress, München, November 2019
- [92] NUSSBAUMER, Prof. Dr. T. ; THALMANN, S. ; JENNI, A. ; KÖDEL, J. : *Planungshandbuch Fernwärme*. Bern : Version 1.2, EnergieSchweiz, Bundesamt für Energie BFE, 2018. – Forschungsbericht. – ISBN 3-90870505-30-4
- [93] OELJEKLAUS, G. : *Thermohydraulische Berechnung vermaschter Fernwärmeheizwassernetze*. Dissertation, Universität-GH-Essen, 1987
- [94] OLSTHOORN, D. ; HAGHIGHAT, F. ; MIRZAEI, P. A.: *Integration of storage and renewable energy into district heating systems: A review of modelling and optimization*. Solar Energy, Nr. 136, S. 49-64, Oktober 2016. – <https://doi.org/10.1016/j.solener.2016.06.054>
- [95] PAPAGEORGIOU, M. ; LEIBOLD, M. ; BUSS, M. : *Optimierung: Statische, dynamische, stochastische Verfahren für die Anwendung*. 4. Ausgabe. Springer Vieweg Berlin, Heidelberg, 2015. – ISBN 978-3-662-46936-1
- [96] PAPAKONSTANTINOY, N. ; SAVOLAINEN, J. ; KOISTINEN, J. ; AIKALA, A. ; VYATKIN, V. : *District heating temperature control algorithm based on short term weather forecast and consumption predictions*. IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA), Berlin (Deutschland), September 2016. – <https://doi.org/10.1109/ETFA.2016.7733748>
- [97] PASZKE, A. ; GROSS, S. ; MASSA, F. ; LERER, A. ; BRADBURY GOOGLE, J. ; CHANAN, G. ; KILLEEN, T. ; LIN, Z. ; GIMELSHEIN, N. ; ANTIGA, L. ; DESMAISON, A. ; XAMLA, A. K. ; YANG, E. ; DEVITO, Z. ; RAISON NABLA, M. ; TEJANI, A. ; CHILAMKURTHY, S. ; AI, Q. ; STEINER, B. ; FACEBOOK, L. F. ; FACEBOOK, J. B. ; CHINTALA, S. : *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver (Kanada), Dezember 2019. – <https://doi.org/10.48550/arXiv.1912.01703>
- [98] PEDREGOSA, F. ; VAROQUAUX, G. ; GRAMFORT, A. ; MICHEL, V. ; THIRION, B. ; GRISEL, O. ; BLONDEL, M. ; PRETTENHOFER, P. ; WEISS, R. ; DUBOURG, V. ; VANDERPLAS, J. ; PASSOS, A. ; COURNAPEAU, D. ; BRUCHER, M. ; PERROT, M. ; DUCHESNAY, É. : *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, Nr. 12 (85), S. 2825-2830, 2011. – <https://doi.org/10.48550/arXiv.1201.0490>

- [99] PFRANG, D. ; DIRNER, S. ; DORSCH, K. ; BOHNSACK, D. ; BEICHEL, K. ; FEINDT, O. ; HEINE, F. ; HÖRBRAND, T. ; LENTSCH, D. ; MEINECKE, M. ; NOLD, K. ; PLOTTEN, M. ; SAVVATIS, A. ; SCHÖLDERLE, F. ; WINTER, T. ; ZOSEDER, K. : *GEOmaRe - Ganzheitlich optimierte und nachhaltige Reservoirerschließung für tiefengeothermische Anlagen im bayerischen Molassebecken*. Geothermische Energie, Heft 97, 2020
- [100] PIERL, D. : *Detektion und örtliche Eingrenzung von Leckagen in Fernwärmenetzen auf Basis eines numerisch- analytischen Fernwärmenetzmodells*. Dissertation, Universität Bremen, Fachbereich 01: Physik/Elektrotechnik, Mai 2022. – <https://doi.org/10.26092/elib/1526>
- [101] PINTELON, R. ; SCHOUKENS, J. : *System Identification: A Frequency Domain Approach*. 2. Ausgabe. John Wiles & Sons, Inc., Hoboken, NJ (USA), 2012. – ISBN 978-0-470-64037-1
- [102] PUTTER, Y. C. C.: *Distributed Control Design for District Heating Networks*. Masterthesis, Delft University of Technology, Faculty of Mechanical, Maritime and Materials Engineering, November 2018. – <http://resolver.tudelft.nl/uuid:1f441413-9ae6-4821-a2d5-929bf0f451bf> – Zugriff: 14.10.2022
- [103] RAMACHANDRAN, P. ; ZOPH, B. ; LE, Q. V.: *Swish: A self-gated activation function*. arXiv, 2017. – <https://doi.org/10.48550/arXiv.1710.05941>
- [104] RENE, A. S.: *Model Predictive Control of District Heating Systems*. Masterthesis, Norwegian University of Science and Technology, Faculty of Information Technology, Mathematics and Electrical Engineering, Juni 2016. – https://ntnuopen.ntnu.no/ntnu-xmlui/bitstream/handle/11250/2404711/14509_FULLTEXT.pdf?sequence=1 – Zugriff: 14.10.2022
- [105] REYNOLDS, J. ; AHMAD, M. W. ; REZGUI, Y. ; HIPPOLYTE, J. L.: *Operational supply and demand optimisation of a multi-vector district energy system using artificial neural networks and a genetic algorithm*. Applied Energy, Nr. 235, S. 699-713, Februar 2019. – <https://doi.org/10.1016/j.apenergy.2018.11.001>
- [106] RUNVIK, H. ; LARSSON, P.-O. ; VELUT, S. ; FUNQUIST, J. ; BOHLIN, M. ; NILSSON, A. ; RAZAVI, S. M.: *Production Planning for Distributed District Heating Networks with JModelica.org*. Proceedings of the 11th International Modelica Conference, Versailles (Frankreich), September 2015. – <http://dx.doi.org/10.3384/ecp15118217>
- [107] SAMETI, M. ; HAGHIGHAT, F. : *Optimization approaches in district heating and cooling thermal network*. Energy & Buildings, Nr. 140, S. 121-130, April 2017. – <https://doi.org/10.1016/j.enbuild.2017.01.062>
- [108] SANDOU, G. ; FONT, S. ; TEBBANI, S. ; HIRET, A. ; MONDON, C. : *Optimization and control of a district heating network*. IFAC Proceedings Volumes, Nr. 38 (1), S. 397-402, 2005. – <https://doi.org/10.3182/20050703-6-CZ-1902.00466>
- [109] SANDOU, G. ; FONT, S. ; TEBBANI, S. ; HIRET, A. ; MONDON, C. : *Predictive Control of a Complex District Heating Network*. Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference, Sevilla (Spanien), S.7372-7377, Dezember 2005. – <https://doi.org/10.1109/CDC.2005.1583351>
- [110] SANDOU, G. ; FONT, S. ; TEBBANI, S. ; HIRET, A. ; MONDON, C. : *Global modelling and simulation of a district heating network*. 9th International Symposium on District Heating and Cooling, Espoo (Finnland), August 2004

- [111] SANDROCK, Dr. M. ; MAASS, C. ; WEISLEDER, S. ; KAUFMANN, C. ; FUSS, G. ; SØRENSEN, P. A. ; LAURBERG JENSEN, L. ; RADMANN, K. : *Erneuerbare Energien im Fernwärmenetz Hamburg, Teil 1: Handlungsoptionen für einen kurzfristigen Ersatz des Kraftwerkes Wedel*. Gutachten, HIC Hamburg Institut Consulting GmbH, Dezember 2016. – https://www.hamburg-institut.com/wp-content/uploads/2021/06/Erneuerbare_Energien_im_Fernwaermenetz_Hamburg.pdf – Zugriff: 14.10.2022
- [112] SCHULZ, H. : *Die Pumpen: Arbeitsweise, Berechnung, Konstruktion*. 13. Auflage. Springer Berlin, Heidelberg, 1977. – ISBN 978-3-642-93054-6
- [113] SCHWAN, T. ; ZIESSLER, O. ; ECKHARDT, T. ; UNGER, R. : *A Modelica-Based Framework for District Heating Grid Simulation*. Proceedings of the 13th International Modelica Conference, Regensburg (Deutschland), S. 159-168, März 2019. – <http://doi.org/10.3384/ecp19157159>
- [114] SCHWEIGER, G. ; LARSSON, P.-O. ; MAGNUSSON, F. ; LAUENBURG, P. ; VELUT, S. : *District heating and cooling systems – Framework for Modelica-based simulation and dynamic optimization*. Energy, Nr. 137, S. 566-578, Oktober 2017. – <https://doi.org/10.1016/j.energy.2017.05.115>
- [115] SIGLOCH, H. : *Technische Fluidmechanik*. 10. Auflage. Springer Vieweg Berlin, Heidelberg, 2017. – ISBN 978-3-662-54467-9
- [116] SOLINAS, F. M. ; BOTTACCIOLI, L. ; GUELPA, E. ; VERDA, V. ; PATTI, E. : *Peak shaving in district heating exploiting reinforcement learning and agent-based modelling*. Engineering Applications of Artificial Intelligence, Nr. 102, 104235, Juni 2021. – <https://doi.org/10.1016/j.engappai.2021.104235>
- [117] STEINDL, G. : *Thermohydraulische Simulation zur Untersuchung des Speicherpotentials von Fernwärmenetzen*. Forschungsforum der Österreichischen Fachhochschulen 2017, 2017. – http://ffhoarep.fh-ooe.at/bitstream/123456789/990/1/Panel_116_ID_178.pdf – Zugriff: 14.10.2022
- [118] STORN, R. ; PRICE, K. : *Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces*. Journal of Global Optimization, Nr. 11, S. 341-359, 1997. – <https://doi.org/10.1023/A:1008202821328>
- [119] STRELOW, O. : *Optimierung von Erdgasversorgungsnetzen*. VDI-Tagung Optimierung in der Energieversorgung, Stuttgart (Deutschland), 1994
- [120] SZEPANSKY, T. ; ZIEMANN, O. : *Verbundprojekt: Bessere Ausnutzung von Fernwärmeanlagen. Teilvorhaben: Optimierung der Wärmeverteilung - Pilotprojekt Berlin*. Schlussbericht Forschungsvorhaben 0326907D/3, Bewag AG, Berlin (Deutschland), September 2001. – <https://www.osti.gov/etdeweb/servlets/purl/20293974> – Zugriff: 14.10.2022
- [121] TALEBI, B. ; MIRZAEI, P. A. ; BASTANI, A. ; HAGHIGHAT, F. : *A Review of District Heating Systems: Modeling and Optimization*. Frontiers in Built Environment, Nr. 2 (16), Oktober 2016. – <http://dx.doi.org/10.3389/fbuil.2016.00022>
- [122] THE MATHWORKS INC.: *MATLAB fmincon (Version 2020b)*. – <https://de.mathworks.com/help/optim/ug/fmincon.html> – Zugriff: 14.10.2022 – [Software]
- [123] THE MATHWORKS INC.: *MATLAB System Identification Toolbox (Version 2020b)*. – <https://de.mathworks.com/help/ident/> – Zugriff: 14.10.2022 – [Software]
- [124] THE SCIPY COMMUNITY: *SciPy: Optimization and root finding (Version 1.7.3)*. – <https://docs.scipy.org/doc/scipy/reference/optimize.html> – Zugriff: 14.10.2022 – [Software]

- [125] TUCHS, M. ; HILMER, W. ; OEL, M. ; SCHAPER, H. ; MICUS, W. ; TEGELER, C. : *Verbundprojekt : Bessere Ausnutzung von Fernwärmeanlagen: Teilprojekt Hannover – Hamburg: „Analyse des Regelverhaltens von Fernwärmenetzen“*. Schlussbericht Forschungsvorhaben 0326993D, Stadtwerke Hannover AG, 2001. – <https://edocs.tib.eu/files/e01fb01/329775480.pdf> – Zugriff: 14.10.2022
- [126] VANDERMEULEN, A. ; HEIJDE, B. van d. ; HELSEN, L. : *Controlling District Heating and Cooling Networks to Unlock Flexibility: a Review*. Energy, Nr. 151, S. 103-115, Mai 2018. – <https://doi.org/10.1016/j.energy.2018.03.034>
- [127] VANDERMEULEN, A. ; HEIJDE, B. van d. ; VANHOUDT, D. ; SALENBIEN, R. ; HELSEN, L. ; VANHOUDT, D. ; SALENBIEN, R. ; HELSEN, L. : *Modesto - a Multi-Objective District Energy Systems Toolbox for Optimization*. The 5th International Solar District Heating Conference, Graz (Österreich), S. 140-147, Mai 2018
- [128] VDI E. V.: *VDI-Wärmeatlas*. 11. Auflage. Springer Vieweg Berlin, Heidelberg, 2013. – ISBN 978-3-642-19981-3
- [129] VERRILLI, F. ; SRINIVASAN, S. ; GAMBINO, G. ; CANELLI, M. ; HIMANKA, M. ; DEL VECCHIO, C. ; SASSO, M. ; GLIELMO, L. : *Model Predictive Control-Based Optimal Operations of District Heating System with Thermal Energy Storage and Flexible Loads*. IEEE Transactions on Automation Science and Engineering, Nr. 14 (2), S. 547-557, November 2016. – <https://doi.org/10.1109/TASE.2016.2618948>
- [130] VESTERLUND, M. ; DAHL, J. : *A method for the simulation and optimization of district heating systems with meshed networks*. Energy Conversion and Management, Nr. 89, S. 555-567, Januar 2015. – <https://doi.org/10.1016/j.enconman.2014.10.002>
- [131] VESTERLUND, M. ; TOFFOLO, A. : *Design Optimization of a District Heating Network Expansion, a Case Study for the Town of Kiruna*. Applied Sciences, Nr. 7 (5), 488, Mai 2017. – <https://doi.org/10.3390/app7050488>
- [132] VESTERLUND, M. ; TOFFOLO, A. ; DAHL, J. : *Simulation and analysis of a meshed district heating network*. Energy Conversion and Management, Nr. 122, S. 63-73, August 2016. – <https://doi.org/10.1016/j.enconman.2016.05.060>
- [133] VESTERLUND, M. ; TOFFOLO, A. ; DAHL, J. : *Optimization of multi-source complex district heating network , a case study*. Energy, Nr. 126, S. 53-63, Mai 2017. – <https://doi.org/10.1016/j.energy.2017.03.018>
- [134] VOLUE ASA: *BoFiT Optimierung*. – <https://www.volue.com/de/product/bofit-optimization> – Zugriff: 14.10.2022 – [Software]
- [135] VOLUE ASA: *BoFiT Prognose*. – <https://www.volue.com/de/product/bofit-forecast> – Zugriff: 14.10.2022 – [Software]
- [136] WANG, H. ; TU, F. ; TU, B. ; FENG, G. ; YUAN, G. ; REN, H. ; DONG, J. : *Neural Network Based Central Heating System Load Prediction and Constrained Control*. Mathematical Problems in Engineering, Februar 2018. – <https://doi.org/10.1155/2018/2908608>
- [137] WIESE, F. ; BRAMSTOFT, R. ; KODUVERE, H. ; PIZARRO ALONSO, A. ; BALYK, O. ; KIRKERUD, J. G. ; TVETEN, Å. G. ; FOLSLAND BOLKESJØ, T. ; MÜNSTER, M. ; RAVN, H. : *Balmorel open source energy system model*. Energy Strategy Reviews, Nr. 20, S. 26-34, April 2018. – <https://doi.org/10.1016/j.esr.2018.01.003>
- [138] WILTSHIRE, R. : *Advanced District Heating and Cooling (DHC) Systems*. 1. Auflage. Woodhead Publishing, 2016. – ISBN 978-1-78242-395-9

- [139] WIRTHS, A. : *Einfluss der Netzrücklauf­temperatur auf die Effizienz von Fernwärmesystemen*. 13. Dresdner Fernwärmekolloquium, Dresden (Deutschland), September 2008. – https://tu-dresden.de/ing/maschinenwesen/iet/gewv/ressourcen/dateien/forschung_und_projekte/projekte/mldh/vortraege/wirths_ruecklauf­temperatur_13_dresdner_fernwaermekolloquium.pdf?lang=de – Zugriff: 14.10.2022
- [140] YABANOVA, I. ; KEÇEBAŞ, A. : *Development of ANN model for geothermal district heating system and a novel PID-based control strategy*. Applied Thermal Engineering, Nr. 51 (1-2), S. 908-916, März 2003. – <https://doi.org/10.1016/j.applthermaleng.2012.10.044>
- [141] ZABALA, L. ; FEBRES, J. ; STERLING, R. ; LÓPEZ, S. ; KEANE, M. : *Virtual testbed for model predictive control development in district cooling systems*. Renewable and Sustainable Energy Reviews, Nr. 129, 109920, September 2020. – <https://doi.org/10.1016/j.rser.2020.109920>
- [142] ZHANG, T. ; LUO, J. ; CHEN, P. ; LIU, J. : *Flow Rate Control in Smart District Heating Systems Using Deep Reinforcement Learning*. arXiv, Dezember 2019. – <https://doi.org/10.48550/arXiv.1912.05313>