

Masterthesis

Vergleichende Analyse von Software zur Modellierung von Energiesystemen mittels Integration in ein Framework zur Transformation

Darwin Schnute

Matr.-Nr.: 21486517

Erstprüferin: Dr.-Ing. Kristin Abel-Günther

Zweitprüfer: Prof. Dr.-Ing. Christian Becker

Betreuer: M.Sc. Mathias Ammon

Hamburg, 31. Januar 2022

Eidesstattliche Erklärung

Ich erkläre hiermit, dass die vorliegende Masterarbeit ohne fremde Hilfe selbstständig verfasst wurde und nur die angegebenen Quellen und Hilfsmittel benutzt worden sind. Wörtlich oder sinngemäß aus anderen Werken entnommene Stellen sind unter Angabe der Quelle kenntlich gemacht. Diese Arbeit ist nach bestem Wissen erstellt worden. Für den Inhalt kann jedoch keine Gewähr übernommen werden.

Hamburg, 31. Januar 2022

(Darwin Schnute)

Abstract

The *Tessif* project, in the context of which this thesis was written, provides a software interface that enables an automated and standardized comparison of software for the modelling and optimization of energy supply systems. So far, *Oemof* and *PyPSA* are included and to which an energy system formulated in *Tessif* can be passed for optimization. The approach is based on two functions individually developed for the software model, to perform pre- and post-processing of all relevant components, parameters and data to realistically describe the energy system. *Tessif* is designed to compare the respective results, optimized with the same basis and to work out distinctive differences. The aim of this thesis is to integrate the software model *FINE* and to examine first results concerning an example energy system. The main task is to develop the individual functions for the automated transformation (*es2es*) as well as the standardized presentation of the results (*es2mapping*). Software for the optimization of energy systems usually makes use of five basic components (source, sink, energy conversion, storage and transmission). That serve to formulate and investigate a wide range of research topics. The intention of the translation is to automatically transform all components available in *Tessif* into the *FINE* specific interface and make them available for optimization. The second function to be developed enables the standardized presentation of the results. Relevant is detailed information about generation and load profiles, installed capacities as well as energy flow related emissions and costs. Finally, the total costs and emissions calculated during the optimization can be used as a basis for testing results of the respective software and making them initially comparable. In addition, a scenario analysis was carried out in this thesis, using an energy system specially developed for the comparison, in order to verify the function and results of *FINE*. Overall, it could be shown that the integration of *FINE* achieves approximately the same results as the integrations of *Oemof* and *PyPSA* and therefore is correctly integrated. Minor differences could be identified, analyzed and, if possible, adjusted to make the software models behave identically. In summary, initial results demonstrated that the capabilities of *Tessif* have been improved and, based on this, a recommendation can be formulated in the future to determine which software achieves the highest benefit in which scientific energy sector.

Kurzfassung

Das Projekt *Tessif*, in dessen Rahmen diese Arbeit verfasst wurde, stellt ein Softwareinterface, das einen automatisierten und standardisierten Vergleich von Software zur Modellierung und Optimierung von Energiesystemen ermöglicht. Bislang enthalten sind *Oemof* und *PyPSA*, an die jeweils ein in *Tessif* formuliertes Energiesystem zur Optimierung übergeben werden kann. Der Ansatz basiert auf zwei für das Softwaremodell individuell entwickelter Funktionen für die Vor- und Nachbereitung aller relevanten Komponenten, Parameter und Daten, um das Energiesystem realistisch zu beschreiben. *Tessif* wurde entwickelt, um die jeweiligen Ergebnisse gegenüberzustellen und markante Unterschiede herauszuarbeiten. Das Ziel dieser Arbeit ist es das Softwaremodell *FINE* zu integrieren und erste Ergebnisse bezüglich eines Beispielenergiesystems zu untersuchen. Die Hauptaufgabe ist die individuellen Funktionen für die automatisierte Übersetzung (*es2es*) sowie die standardisierte Darstellung der Ergebnisse (*es2mapping*) zu entwickeln. Die Intention der Übersetzung ist es alle in *Tessif* vorhandenen Komponenten in *FINE* zu transformieren und der Optimierung zur Verfügung zu stellen. Die zweite zu entwickelnde Funktion ermöglicht die standardisierte Darstellung der Ergebnisse. Relevant sind detaillierte Informationen zu Erzeugungs- und Lastprofilen, den installierten Leistungen bzw. Kapazitäten sowie den flussbezogenen Emissionen und Kosten. Schließlich kann durch die während der Optimierung berechneten Gesamtkosten- und Emissionen eine Basis geschaffen werden, um die Ergebnisse der jeweiligen Software zu prüfen und vergleichbar zu machen. Mit Hilfe eines speziell für den Vergleich entwickelten Energiesystems ist eine Szenarioanalyse durchgeführt worden, um die Funktion und die Ergebnisse von *FINE* zu prüfen. Insgesamt konnte gezeigt werden, dass die Integration von *FINE* annähernd gleiche Ergebnisse erzielt wie *Oemof* und *PyPSA* und damit korrekt integriert wurde. Geringfügige Unterschiede konnten identifiziert, analysiert und, sofern möglich, angepasst werden, damit die Softwaremodelle gleiches Verhalten aufweisen. Zusammenfassend konnte anhand erster Ergebnisse demonstriert werden, dass die Fähigkeiten von *Tessif* erweitert wurden und zukünftig auf dieser Grundlage eine Empfehlung formuliert werden kann, welche Software für die unterschiedlichen energietechnischen Bereiche den höchsten Nutzen erzielt.

Inhaltsverzeichnis

Abstract	I
Kurzfassung	III
Inhaltsverzeichnis	V
Abbildungsverzeichnis	VII
Tabellenverzeichnis	IX
Abkürzungsverzeichnis	XII
Formelzeichenverzeichnis	XIII
1 Motivation	1
2 Theorie	3
2.1 Grundlagen der Modellierung von Energiesystemen.....	3
2.2 Anwendungen der Modellierung.....	7
2.3 Simulation und Optimierung.....	8
2.4 Mathematische Grundlagen.....	9
2.4.1 Lineare Programmierung (LP).....	10
2.4.2 Gemischt-ganzzahlige lineare Programmierung (MILP).....	13
2.5 Vergleich von Software zur Modellierung und Optimierung.....	14
2.5.1 Stand der Forschung.....	15
2.5.2 Transforming Energy Supply System Frameworks (<i>Tessif</i>).....	18
2.6 Framework for Integrated Energy System Assessment (<i>FINE</i>).....	20
2.6.1 Komponenten.....	21
2.6.2 Fähigkeiten und Grenzen.....	22
2.6.3 Time Series Aggregation Module (<i>tsam</i>).....	23
3 Methodik	25
3.1 Auswahl des Softwaremodells.....	26
3.2 Integration des Softwaremodells.....	28
3.2.1 Energy-System to Energy-System (<i>es2es</i>).....	29
3.2.2 Energy-System to Mapping (<i>es2mapping</i>).....	42
3.3 Auswahl und Beschreibung des Beispielenergiesystems.....	49
4 Auswertung und Vergleich	55
4.1 Aggregation der zeitlichen Auflösung.....	56
4.2 Szenarioanalyse.....	63

4.2.1 Vergleich der Optimierungszeiten	64
4.2.2 Referenzszenario	66
4.2.3 Basisszenario.....	77
5 Zusammenfassung und Fazit	87
5.1 Zusammenfassung.....	87
5.2 Fazit	89
6 Ausblick.....	91
Literaturverzeichnis.....	XV
Anhang.....	XIX

Abbildungsverzeichnis

Abbildung 2-1: Schema des Vorgehens einer Software zur Modellierung von Energiesystemen.....	4
Abbildung 2-2: Abstrakte Darstellung eines vereinfachten Energiesystems.	5
Abbildung 2-3: Abstrakte Darstellung eines Energiesystems mit zwei Energieträgern, einer Energieumwandlung, zugehörigen Speichern, Ersatzquellen und Transport.....	6
Abbildung 2-4: Zulässiger Bereich einer linearen Optimierung zur Minimierung bezüglich der Aktivität.	11
Abbildung 2-5: Schematische Darstellung des Simplex-Verfahrens und der Inneren-Punkt Methode für ein zu minimierendes System bezüglich der Aktivität x	12
Abbildung 2-6: Zustandsbaum als Grundlage für die Branch-and-bound-Methode.	14
Abbildung 2-7: Überschneidung der Ergebnisse von Softwaremodellen im gleichen Lösungsraum.	15
Abbildung 2-8: Schematische Darstellung der Einsatzweise von <i>Tessif</i> mit den Funktionen für die Übersetzung der Energiesysteme (<i>es2es</i>) und für die standardisierte Darstellung der Ergebnisse (<i>es2mapping</i>).....	18
Abbildung 2-9: Graphische Darstellung des Energiesystems aus Abbildung 2-3 visualisiert mittels <i>Tessif</i>	19
Abbildung 3-1: Schematische Darstellung der Einsatzweise von <i>Tessif</i> , erweitert mit dem Ziel der Arbeit.	25
Abbildung 3-2: Schematische Darstellung der Transformation der Komponenten und zugehörigen Parametern von <i>Tessif</i> zu <i>FINE</i>	30
Abbildung 3-3: Schematische Darstellung des Konnektors in <i>Tessif</i> entwickelt für <i>FINE</i> durch Kombination von Energieumwandlungen.	37
Abbildung 3-4: Schematische Darstellung der emittierenden Quelle entwickelt für <i>FINE</i> durch Kombination einer fiktiven Quelle und einer Energieumwandlung.	38
Abbildung 3-5: Schematische Darstellung eines emittierenden Speichers als indirekte Komponente entwickelt für <i>FINE</i> durch Kombination mit Energieumwandlungen.	39
Abbildung 3-6: Visualisierung der Modellarchitektur des <i>component_es</i> der <i>Tessif</i> -Bibliothek.....	52

Abbildung 4-1: Zusammensetzung des Stromsektors optimiert durch <i>FINE</i> mit und ohne aggregierter Datengrundlage durch <i>tsam</i> für die Szenarien des <i>component_es</i> der <i>Tessif</i> -Bibliothek.....	58
Abbildung 4-2: Zusammensetzung des Wärmesektors optimiert durch <i>FINE</i> mit und ohne aggregierter Datengrundlage durch <i>tsam</i> für die Szenarien des <i>component_es</i> der <i>Tessif</i> -Bibliothek.....	60
Abbildung 4-3: Jahresdauerlinien der On- und Offshore-WKA optimiert durch <i>FINE</i> mit und ohne aggregierter Datengrundlage durch <i>tsam</i> für die Szenarien des <i>component_es</i> der <i>Tessif</i> -Bibliothek.....	62
Abbildung 4-4: Zusammensetzung des Stromsektors für das Referenzszenario des <i>component_es</i> der <i>Tessif</i> -Bibliothek optimiert durch <i>FINE</i> , <i>Oemof</i> und <i>PyPSA</i>	67
Abbildung 4-5: Erzeugungsprofil des Stromsektors eines Wintermonats für das Referenzszenario des <i>component_es</i> der <i>Tessif</i> -Bibliothek optimiert durch <i>FINE</i>	69
Abbildung 4-6: Erzeugungsprofil des Stromsektors eines Sommermonats für das Referenzszenario des <i>component_es</i> der <i>Tessif</i> -Bibliothek optimiert durch <i>FINE</i>	70
Abbildung 4-7: Zusammensetzung des Wärmesektors für das Referenzszenario des <i>component_es</i> der <i>Tessif</i> -Bibliothek optimiert durch <i>FINE</i> , <i>Oemof</i> und <i>PyPSA</i>	71
Abbildung 4-8: Erzeugungsprofil des Wärmesektors eines Wintermonats für das Referenzszenario des <i>component_es</i> der <i>Tessif</i> -Bibliothek optimiert durch <i>FINE</i>	72
Abbildung 4-9: Ladezustand des Wärmespeichers für das Referenzszenario des <i>component_es</i> der <i>Tessif</i> -Bibliothek für das Ende der untersuchten Zeitperiode optimiert durch <i>FINE</i> , <i>Oemof</i> und <i>PyPSA</i>	75
Abbildung 4-10: Zusammensetzung des Stromsektors für das Basisszenario des <i>component_es</i> der <i>Tessif</i> -Bibliothek optimiert durch <i>FINE</i> , <i>Oemof</i> und <i>PyPSA</i>	79
Abbildung 4-11: Erzeugungsprofil des Stromsektors eines Wintermonats für das Basisszenario des <i>component_es</i> der <i>Tessif</i> -Bibliothek optimiert durch <i>FINE</i>	81
Abbildung 4-12: Erzeugungsprofil des Stromsektors eines Sommermonats für das Basisszenario des <i>component_es</i> der <i>Tessif</i> -Bibliothek optimiert durch <i>FINE</i>	82
Abbildung 4-13: Zusammensetzung des Wärmesektors für das Basisszenario des <i>component_es</i> der <i>Tessif</i> -Bibliothek optimiert durch <i>FINE</i> , <i>Oemof</i> und <i>PyPSA</i>	83
Abbildung 4-14: Erzeugungsprofil des Wärmesektors eines Wintermonats für das Basisszenario des <i>component_es</i> der <i>Tessif</i> -Bibliothek optimiert durch <i>FINE</i>	84

Tabellenverzeichnis

Tabelle 3-1: Übersicht der Auswahlkriterien für in <i>Tessif</i> zu integrierende Software.	27
Tabelle 3-2: <i>Unique Identifier</i> für die eindeutige Beschreibung einer Komponente des Energiesystems.	28
Tabelle 3-3: Energiesystemparameter durch die Übersetzungsfunktion in <i>Tessif</i>	29
Tabelle 3-4: Bezeichnung der Komponenten in <i>Tessif</i> und <i>FINE</i>	30
Tabelle 3-5: Spezifische Parameter des Energietransportes für <i>Tessif</i> und <i>FINE</i>	32
Tabelle 3-6: Spezifische Parameter der Senke für <i>Tessif</i> und <i>FINE</i>	32
Tabelle 3-7: Spezifische Parameter der Quelle für <i>Tessif</i> und <i>FINE</i>	33
Tabelle 3-8: Spezifische Parameter der Energieumwandlung für <i>Tessif</i> und <i>FINE</i>	34
Tabelle 3-9: Spezifische Parameter des Speichers für <i>Tessif</i> und <i>FINE</i>	35
Tabelle 3-10: Spezifische Parameter der Grenzwerte in <i>Tessif</i> und <i>FINE</i>	36
Tabelle 3-11: Spezifische Parameter des Konnektors in <i>Tessif</i> entwickelt für <i>FINE</i> durch Kombination von Energieumwandlungen.	37
Tabelle 3-12: Spezifische Parameter der emittierenden Quelle entwickelt für <i>FINE</i> durch Kombination einer fiktiven Quelle und einer Energieumwandlung.	39
Tabelle 3-13: Spezifische Parameter eines emittierenden Speichers als indirekte Komponente entwickelt für <i>FINE</i> durch Kombination mit Energieumwandlungen.	40
Tabelle 3-14: Beispiel Energiesysteme für Basiskomponenten in <i>Tessif</i> und <i>FINE</i>	41
Tabelle 3-15: Vorzeichendefinition eintretende Ströme negativ, austretende positiv einer Energieumwandlung über drei Zeitschritte.	43
Tabelle 3-16: <i>FINE</i> spezifische Parameter Dimension, Vorzeichen und Energieträger für die eindeutige Beschreibung der Komponenten des in Abbildung 2-3 und Abbildung 2-9 dargestellten Energiesystems.	44
Tabelle 3-17: Spezifikationen der Szenarien des <i>component_es</i> der <i>Tessif</i> Bibliothek.	49
Tabelle 3-18: Strom- und Wärmebedarf für das <i>component_es</i> der <i>Tessif</i> -Bibliothek.	50
Tabelle 3-19: Technische Parameter der Komponenten des <i>component_es</i> der <i>Tessif</i> Bibliothek.	53
Tabelle 3-20: Ökonomische Parameter der Komponenten des <i>component_es</i> der <i>Tessif</i> Bibliothek.	53

Tabelle 4-1: Mittlere Berechnungszeiten der Optimierung durch *FINE* mit und ohne aggregierter Datengrundlage durch *tsam* für die Szenarien des *component_es* der *Tessif*-Bibliothek.....56

Tabelle 4-2: Gesamtergebnisse zu Kosten und Emissionen optimiert durch *FINE* mit und ohne aggregierter Datengrundlage durch *tsam* für die Szenarien des *component_es* der *Tessif*-Bibliothek.....57

Tabelle 4-3: Relative Abweichung der bereitgestellten Strommengen von den Ergebnissen mit aggregierter Datengrundlage durch *tsam* zu denen ohne Aggregation. Optimiert durch *FINE* für die Szenarien des *component_es* der *Tessif*-Bibliothek.59

Tabelle 4-4: Relative Abweichung der bereitgestellten Wärmemengen von den Ergebnissen mit aggregierter Datengrundlage durch *tsam* zu denen ohne Aggregation. Optimiert durch *FINE* für die Szenarien des *component_es* der *Tessif*-Bibliothek.60

Tabelle 4-5: Installierte Leistungen bzw. Kapazitäten optimiert durch *FINE* mit und ohne aggregierter Datengrundlage durch *tsam* für das Basisszenario des *component_es* der *Tessif*-Bibliothek.....61

Tabelle 4-6: Mittlere Berechnungszeiten der Optimierung durch *Oemof*, *PyPSA* und *FINE* für die Szenarien des *component_es* der *Tessif*-Bibliothek.65

Tabelle 4-7: Gesamtergebnisse zu Kosten und Emissionen für das Referenzszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *FINE*, *Oemof* und *PyPSA*.66

Tabelle 4-8: Abweichung der bereitgestellten Strommengen von *Oemof* und *PyPSA* relativ zu *FINE* für das Referenzszenario des *component_es* der *Tessif*-Bibliothek.68

Tabelle 4-9: Abweichung der bereitgestellten Wärmemengen von *Oemof* und *PyPSA* relativ zu *FINE* für das Referenzszenario des *component_es* der *Tessif*-Bibliothek.73

Tabelle 4-10: Speicherdissipation durch Selbstentladung in Folge der Optimierung durch *FINE*, *Oemof* und *PyPSA* für das Referenzszenario des *component_es*74

Tabelle 4-11: Wärmespeicherstand für den ersten und letzten Zeitschritt in Folge der Optimierung durch *FINE*, *Oemof* und *PyPSA* für das Referenzszenario des *component_es* der *Tessif*-Bibliothek.....75

Tabelle 4-12: Gesamtergebnisse zu Kosten und Emissionen für das Basisszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *FINE*, *Oemof* und *PyPSA*.77

Tabelle 4-13: Installierte Leistungen bzw. Kapazitäten optimiert durch *FINE*, *Oemof* und *PyPSA* für das Basisszenario des *component_es* der *Tessif*-Bibliothek. Relative Abweichung zu *FINE*78

Tabelle 4-14: Abweichung der bereitgestellten Strommengen von *Oemof* und *PyPSA* relativ zu *FINE* für das Basisszenario des *component_es* der *Tessif*-Bibliothek. 80

Tabelle 4-15: Abweichung der bereitgestellten Wärmemengen von *Oemof* und *PyPSA* relativ zu *FINE* für das Basisszenario des *component_es* der *Tessif*-Bibliothek. 83

Tabelle 4-16: Gesamtergebnisse zu Kosten und Emissionen unter Vernachlässigung der spezifischen Emissionen von Batterie und PtH-Anlage für das Basisszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *FINE*, *Oemof* und *PyPSA*. 85

Abkürzungsverzeichnis

BHKW	Blockheizkraftwerk
BK	Braunkohle
CAPEX	Investitionskosten
CBC	Coin-Or Branch Solver
com	Energieträger (Commodity)
CO ₂	Kohlenstoffdioxid
dim	Dimension einer Komponente
ES	Energiesystem
<i>es2es</i>	Energy-system to energy-system
<i>es2mapping</i>	Energy-system to mapping
<i>FINE</i>	Framework for Integrated Energy System Assessment
GLPK	GNU Linear Programming Kit
PtH	Power-to-Heat
KW	Kraftwerk
KWK	Kraft-Wärme Kopplung
inst.	Installiert
LP	Lineare Programmierung
MILP	Gemischt-ganzzahlige lineare Programmierung
<i>Oemof</i>	Open energy system modelling framework
OPEX	Betriebskosten
PV	Photovoltaik
<i>PyPSA</i>	Python for Power System Analysis
RES	Renewable Energy Sources (Quellen erneuerbarer Energien)
SK	Steinkohle
sign	Parameter der Quelle (+1) oder Senke (-1) in <i>FINE</i>
SOC	Ladestand (State of Charge)
spez.	Spezifisch
<i>Tessif</i>	Transforming Energy Supply System (Modelling) Frameworks
UID	Unique Identifier für Eindeutigkeit der Komponenten
WKA	Windkraftanlage

Formelzeichenverzeichnis

A	System aus Gleichungen und Ungleichungen
b	Nebenbedingung
c	Lineare Zielfunktion
$CAPEX$	Investitionskosten
$CAPEX_{ges}$	Gesamte Investitionskosten berechnet durch <i>Tessif</i>
$CAPEX_{inst.}$	Investitionskosten der bereits installierten Komponenten
E	Energiemenge
$E_{Batterie}$	Energiemenge bereitgestellt durch die Batterie
E_{PtH}	Energiemenge bereitgestellt durch die PtH-Anlage
E_W	Energiemenge aus Wärmespeicher entladen
$E_{W,Entladen}$	Energiemenge aus Wärmespeicher entladen
$E_{W,Laden}$	Energiemenge in Wärmespeicher geladen
e	Gesamte Emissionen einer Komponente
$Kosten_{ges}$	Gesamte Kosten berechnet durch das Softwaremodell
n	Anzahl aller Komponenten im Energiesystem
$OPEX$	Betriebskosten
$OPEX_{ges}$	Gesamte Betriebskosten berechnet durch <i>Tessif</i>
P	Zulässiger Bereich
P_{0-4}	Teilsysteme im Branch-and-Bound Verfahren
$P_{K,ges}$	Gesamte bereitgestellte Energiemenge einer Komponente
x	Entscheidungsvariable, Aktivität
$\Delta E_{W,FINE}$	Differenz der Dissipation des Wärmespeichers
η	Wirkungsgrad
η_{WS}	Wirkungsgrad des Wärmespeichers

1 Motivation

Die Datenstrategie der Bundesregierung sieht die effiziente Nutzung von Energiedaten vor, welche durch die digitale Transformation des Energiesektors zur Verfügung stehen. Mit Hilfe solcher Daten kann das Energiesystem der Zukunft gestaltet und die Kopplung der Sektoren vorangetrieben werden [1]. Um die politischen Emissionsziele für eine Klimaneutralität einzuhalten, gilt es die Fluktuation erneuerbarer Energien auszugleichen, damit wirtschaftlich und gleichzeitig ökologisch Energie in jeder Form bereitgestellt werden kann. Eine Schlüsselaufgabe ist aufgrund dessen die Dezentralisierung des Energiesystems und die Realisierung von Smart Grids. Ziel ist es, das Energiesystem zu flexibilisieren, indem Erzeugung und Bedarf in Echtzeit aneinander angepasst werden und Speichertechnologien überschüssige Energie für spätere Zeitpunkte zwischenlagern [2]. Da neben der Energienachfrage auch die Komplexität der gesamten Versorgungsinfrastruktur steigt, erfährt die computergestützte Analyse des Energiesektors gegenwärtig vermehrt wissenschaftliche Beachtung. Softwares für die Modellierung und Optimierung von Energiesystemen in diversen Maßstäben werden entwickelt und angewandt, um zukunftsfähige Lösungen bereitzustellen. Aufgrund der erzielten Ergebnisse können Aussagen und Empfehlungen dazu getroffen werden, welcher technologischer Ausbau erforderlich ist, um spezifische politische oder gesellschaftliche Ziele einzuhalten [3]. Solche Softwares weisen dennoch verschiedene Schwerpunkte, Stärken und Schwächen auf, weshalb ein strukturierter Vergleich ebendieser erforderlich ist [4]. Eine vielversprechende Methode für einen solchen Vergleich ist die automatisierte Verwendung der Softwares mit gleichzeitig standardisierter Darstellung relevanter Optimierungsergebnisse [5].

Insbesondere im Hinblick auf die erforderliche Energiewende ist es essenziell, die bereits entwickelten Softwaremodelle zu analysieren und zu vergleichen. Mit der Hilfe eines Vergleichs kann eine Empfehlung ausgesprochen werden, in welchen Anwendungsgebieten der größtmögliche Nutzen erzielt werden kann [6].

2 Theorie

Im Folgenden werden die theoretischen Grundlagen für die Modellierung und Optimierung von Energiesystemen (ES) erläutert. Damit wird die Auswahl eines geeigneten Softwaremodells ermöglicht, welches schließlich in ein Framework zur Transformation integriert wird. Weiterhin soll eine Grundlage für den anschließenden Vergleich und die Analyse der durch verschiedenen Softwares erzielten Ergebnisse geschaffen werden.

2.1 Grundlagen der Modellierung von Energiesystemen

Ein Modell (lat. Maß, Maßstab) wird allgemein als Abbild der Realität verstanden und soll diese so exakt wie erforderlich darstellen. Für das ES werden Modelle verwendet, um verschiedene Transformationspfade zu untersuchen. Mit Hilfe dieser Informationen können Empfehlungen ermittelt werden, durch welche Veränderungen ein vorgegebenes Ziel erreicht werden kann. In einem Modell sollen ausschließlich relevante Information enthalten sein, die logisch zueinander passen und das ES nicht verkomplizieren. Grundsätzlich soll ermöglicht werden Forschungsthemen zu untersuchen, deren Umsetzung sich als sehr zeit- und kostenintensiv erweisen würde. Zudem ermöglichen es Modelle, Teile eines sehr komplexen Systems einzeln zu untersuchen, um sie später in einen Gesamtkontext zu bringen [7].

Der Ausgangspunkt für eine Modellierung von ES ist die genaue Definition des übergeordneten Ziels, dass bei allen weiteren Prozessentscheidungen verfolgt wird. Danach können Komponenten, die Auswirkungen auf das Ziel haben, definiert und zusammengestellt werden. Als Komponente wird ein Akteur in einem ES bezeichnet, der Einfluss auf die Bereitstellung, den Transport oder den Verbrauch von Energie aufweist. Von Relevanz zur Beschreibung sind technische, ökologische und ökonomische Parameter jeder Komponente. Das gesamte ES muss dabei immer klar von der Systemumgebung abgegrenzt sein, um unerwünschte Wechselwirkungen auszuschließen. Da eine simultane Betrachtung aller Abhängigkeiten und Parameter für das Gesamtsystem oft nicht möglich ist, müssen diese auf die Maßgeblichen eingegrenzt werden [8].

In Abbildung 2-1 ist schematisch das Vorgehen bei der Modellierung eines ES mit Hilfe einer Software dargestellt.

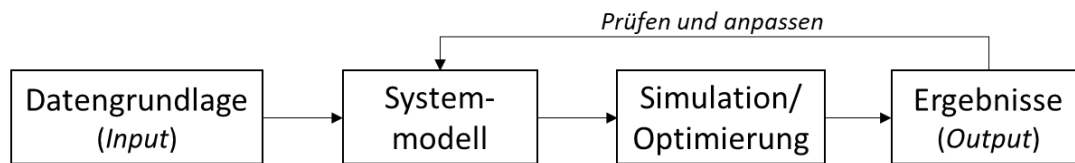


Abbildung 2-1: Schema des Vorgehens einer Software zur Modellierung von Energiesystemen.

Die Datengrundlage (*Input*) besteht aus Wetterdaten oder Erzeugungs- und Verbrauchsprofilen. Weiterhin sind Informationen zu den Komponenten enthalten, die die verwendete Technik beschreiben. Die Summe aller Komponenten repräsentiert dann das entstandene Systemmodell. Dieses beinhaltet alle relevanten Daten und wird an die folgende Berechnung durch eine computergestützte Simulation oder Optimierung übergeben. Ausgegeben werden dann die Ergebnisse in Form von Systemeigenschaften (*Output*). Diese beschreiben die entstehende Realität, wenn die vorab definierten Zielsetzungen verfolgt werden. Die Ergebnisse gilt es im Nachhinein auf Plausibilität zu prüfen und ob diese in einer angemessenen Zeit erzielt werden. Ist dies nicht der Fall, muss das Systemmodell überarbeitet werden, um die Komplexität zu verringern oder Logikfehler zu verbessern [9].

Modellierte ES werden nach ihrem Fokus auf den Verbrauch (*top-down*) oder auf die Versorgung (*bottom-up*) unterschieden. Nach Farzaneh et al. [10] verfolgt die *top-down* Annäherung eine ökonomische Zielsetzung und betrachtet das gesamte ES unter Vernachlässigung von Details. Anwendung finden diese bei der Vorhersage von der Energienachfrage, um die Netzstabilität zu gewährleisten. Um den Verbrauch der Erzeugung anzupassen, könnte auf Basis der erzielten Erkenntnisse beispielsweise eine Lastverschiebung erfolgen. Dagegen soll das *bottom-up* Modell die Betrachtung von einem oder mehreren Teilen des ES detaillierter untersuchen. Maßgeblich beeinflussend sind die techno- ökonomischen Parameter der enthaltenen Komponenten. Bewertet und eingesetzt werden Technologien ausschließlich anhand ihrer Effizienz. Dadurch werden auch Komponenten verwendet, die dem eigentlichen Verbrauchsverhalten widersprechen. Farzaneh et al. beschreiben daher, dass bei dem ökonomischen *top-down* Ansatz eine pessimistische und

bei der ingenieurstechnischen *bottom-up* Betrachtung eine optimistische Zielfindung erfolgt [10].

Pfenniger et al. beschreiben ein Modell des ES als Zusammenspiel zwischen Erzeugung und Verbrauch bei gleichzeitiger Berücksichtigung von Umwandlungs- und Speichertechnologien. Die Verbindung zwischen diesen Komponenten stellt dann der Energietransport als Übertragungs- und Verteilungskomponente dar [11].

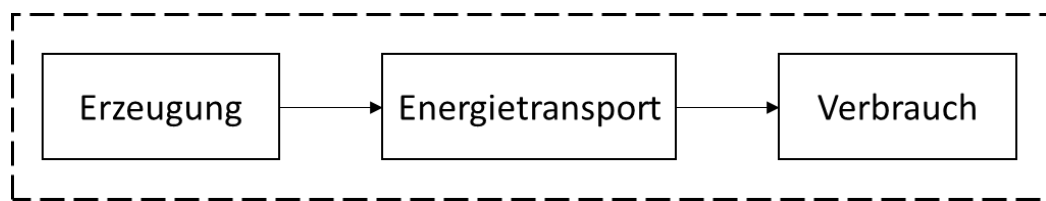


Abbildung 2-2: Abstrakte Darstellung eines vereinfachten Energiesystems.

In Abbildung 2-2 ist ein vereinfachtes ES dargestellt, das aus Erzeugung, Verbrauch und deren Verbindung besteht. Übertragen wird eine bestimmte Energieform innerhalb des Bilanzraums. Anhand der Forderung nach Netzstabilität muss zu jedem Zeitpunkt gewährleistet sein, dass Erzeugung und Bedarf aneinander angepasst sind. Zusätzlich können Komponenten zur Umwandlung und Speicherung genutzt werden, um das ES zu flexibilisieren und mit weiteren Energieflüssen zu erweitern. Allgemein bezeichnet werden Erzeugung, Verbrauch, Energieumwandlung, Speicher und Transport als Basiskomponenten für ein ES. Je detaillierter das System formuliert wird, umso mehr Abhängigkeiten müssen berücksichtigt werden. Wird Abbildung 2-2 um zusätzliche Komponenten erweitert, nimmt die Komplexität des Systems zu.

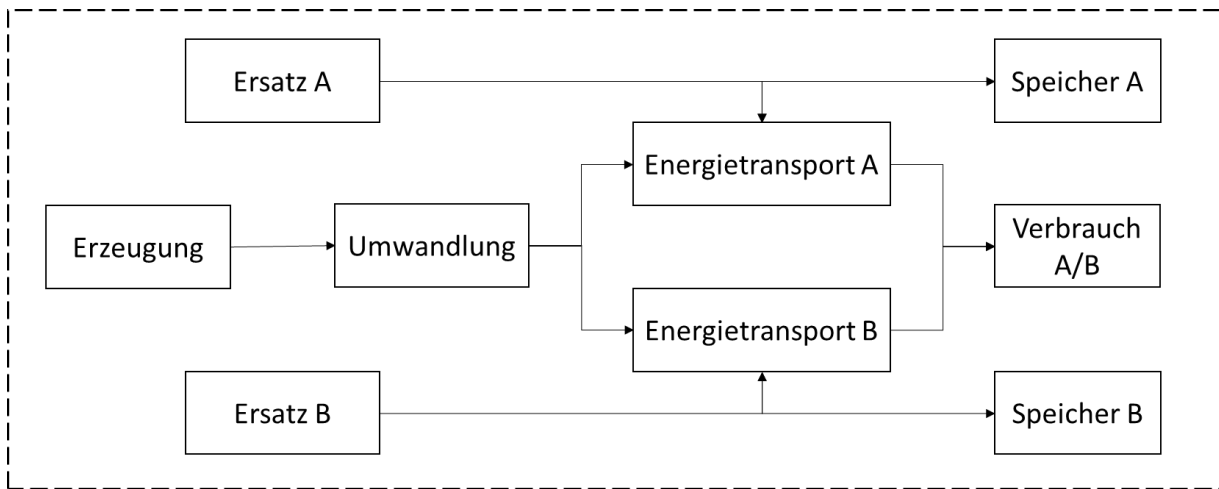


Abbildung 2-3: Abstrakte Darstellung eines Energiesystems mit zwei Energieträgern A und B, einer Energieumwandlung, zugehörigen Speichern, Ersatzquellen und Transport.

Das ES in Abbildung 2-3 besteht aus einer Komponente zur Energieumwandlung, die mit einem Energieträger versorgt wird. Erzeugt werden zwei weitere Energieträger A und B, um den jeweiligen Verbrauch zu decken. Weitergeleitet werden können diese über den jeweiligen Energietransport entweder an den passenden Speicher oder an den Verbrauch. Sollte es nicht möglich sein, den Verbrauch mittels der Umwandlung zu decken, kann auf die Erzeugung durch Ersatzkomponenten zurückgegriffen werden, um Netzstabilität zu gewährleisten. Technisch berücksichtigt werden muss folglich der Wirkungsgrad der Umwandlung sowie die Speicher Lade- und Entladewirkungsgrade. Zusätzlich muss bezüglich des Speichers zumeist eine Selbstentladung berücksichtigt werden, um zwischen Kurz- und Langzeitspeicher unterscheiden zu können. Hauptbedingung für das ES ist die Versorgungssicherheit bei Kostenminimierung und mögliche Nebenbedingungen wären Restriktionen bezüglich Ressourcen und Emissionen. Wird die Systemgrenze dann sukzessive erweitert, zeigt sich, dass die Anzahl an Komponenten und ihrer Abhängigkeiten ausschließlich rechnergestützt mit Hilfe von Softwaremodellen untersucht werden kann. Verwendet werden können die Ergebnisse der Modellierung beispielsweise, um den Entscheidungsprozess für gesellschaftliche oder politische Vorgaben zu definieren bzw. zu unterstützen [10].

2.2 Anwendungen der Modellierung

Hourcade et al. [12] beschreiben drei Anwendungsfelder der Modellierung. Prognosemodelle (*forecasting*) sollen dafür verwendet werden, Vorhersagen über die Zukunft treffen zu können, welche mittels Extrapolation kurzfristige Auswirkungen durch politische oder technische Veränderungen untersuchen. Extrapolation bedeutet in diesem Kontext die strikte Weiterführung zuvor erhobener Daten in die zeitnahe Zukunft. Verwendung finden Prognosemodelle vorwiegend in der Wirtschaft und dienen dazu das Marktverhalten zu analysieren. Im Gegensatz dazu kann mit Rückblickmodellen (*backcasting*) die Einschätzung von Experten verwendet werden, um ein realistisches Zukunftssystem zu entwerfen. Dazu werden vorab Ziele definiert, die dann durch politische oder technische Veränderungen erreicht werden müssen. Verglichen wird das entstandene Modellsystem schließlich mit der Gegenwart, um abzuschätzen, wie die aktuelle Situation verändert werden muss, um das Zukunftssystem umzusetzen. Der dritte Anwendungsbereich ist die Entwicklung von Szenarien, um zwei mögliche Entwicklungspfade des ES direkt miteinander vergleichen zu können. Der Unterschied zwischen einem ES und einem Szenario ist, dass das System die enthaltenen Komponenten und Parameter beschreibt. Das Szenario hingegen formuliert Verhaltensweisen dieser im System vorhandenen Komponenten, um verschiedene Zielsetzungen zu erreichen. Eine Szenarioanalyse erfolgt in der Regel anhand eines Referenz- und eines Basisszenarios [12].

Das Referenzszenario beschreibt eine Zukunft, in der politische Vorgaben oder andere Restriktionen nicht eingehalten werden. Somit resultiert ein realistisches Ergebnis für den Fall, dass keine Veränderungen am ES vorgenommen werden würden. Weiterhin wird in Referenzszenarien davon ausgegangen, dass momentan beeinflussende Bestimmungen, zum Beispiel Subventionen für bestimmte Technologien, nach dem Auslaufen nicht erneut beschlossen werden. Die Erkenntnisse über die entstandenen Kosten und Emissionen dienen als Grundlage für einen Vergleich zu den Ergebnissen des Basisszenarios. Auch als Zielszenario bezeichnet, beschreibt es die notwendige Entwicklung des ES, um politische Vorgaben und Zielsetzungen einzuhalten, bei gleichzeitiger Minimierung der anfallenden Kosten. Dies kann mittels des stetigen Anpassens des ES durch Experten geschehen oder durch eine Software für die Optimierung [13].

Zusammenfassend beschreibt ein typisches Referenzszenario die Einsatzplanoptimierung der bereits im ES vorhandenen Komponenten. Demnach ist es möglich abzuschätzen, inwiefern ohne eine Änderung die Restriktionen verfehlt werden. Das Basisszenario lässt dagegen einen Ausbau zu und zeigt die erforderliche Entwicklung auf, damit die Vorgaben möglichst wirtschaftlich eingehalten werden. Dennoch kann die Szenarioanalyse lediglich die zuvor beschriebene Möglichkeit zur Transformation des ES untersuchen. Für weitere Pfade müssen dann zusätzliche Szenarien entwickelt und miteinander verglichen werden. Verwendung findet die Szenarioanalyse vermehrt während der Modellierung und Optimierung von ES, um die Auswirkung des Ausbaus von erneuerbaren Energien zu untersuchen.

2.3 Simulation und Optimierung

Der Energiesektor kann in seiner Gesamtheit nicht realitätsnah modelliert werden, da die Simulation bzw. Optimierung aufgrund der Komplexität des Systems und der Datengrundlage nicht praktikabel wäre. Daher sollte ein ES so einfach wie möglich und gleichzeitig so komplex wie nötig erstellt werden.

Ist ein System mit allen relevanten Komponenten und zugehörigen Parametern definiert, kann mittels Software eine Simulation durchgeführt werden. Grundsätzlich wird durch eine Simulation das Verhalten der Akteure im untersuchten System für eine gewisse Zeitspanne möglichst realitätsnah beschrieben. Durch Variation von bestimmten Parametern, kann dann überprüft werden, welchen Einfluss diese auf das Gesamtergebnis haben. Dadurch ist es möglich, verschiedene Szenarien und ihre Transformationspfade computergestützt zu untersuchen und anschließend einen Vergleich durchzuführen.

Eine Optimierung kann als Minimierung bzw. Maximierung beschrieben werden, um den aktuellen Zustand eines ES zu verbessern. Vorhanden sind hier Freiheitsgrade, die sukzessive angepasst werden, bis dieser optimale Zustand erreicht ist. Unterschiedliche Zusammensetzungen der Parameter erzeugen verschiedene Ergebnisse der Berechnung. Es entstehen Gleichungssysteme aus Gleichungen und Ungleichungen, die mit Hilfe einer Zielfunktion untersucht werden, sodass die optimale Lösung hinsichtlich des übergeordneten Ziels ermittelt werden kann. Das Gleichungssystem wird dafür an eine Software

zum Lösen (*Solver*) übergeben, die mit Hilfe von mathematischen Methoden über einen Algorithmus das Ergebnis ermittelt [10].

Klassifiziert werden Optimierungsmodelle hinsichtlich ihrer Fähigkeit, zukünftige Gegebenheiten zu berücksichtigen. Bei dem Modell der perfekten Voraussicht sind jegliche Daten der Zeitspanne bekannt, die Einfluss nehmen. So können Zustände am Ende einer Periode den Anfang dieser beeinflussen. Der myopische Ansatz hingegen vermeidet die Verfügbarkeit von solchen Informationen. Unterteilt wird in einzelne Zeitabschnitt, die aufeinander folgend betrachtet werden, um je die optimale Lösung zu ermitteln. Bei dem Ansatz des rollierenden Horizonts werden zusätzlich Parameter jeder Periode an die folgende für die Berechnung übergeben. Somit werden zusätzlich Entwicklungen während der Optimierung berücksichtigt und die Berechnungsdauer wird verringert. Durch den Ansatz des myopischen bzw. rollierenden Horizonts wird gegenüber dem der perfekten Voraussicht ein realitätsnäheres Ergebnis erzielt, da insbesondere fluktuierende erneuerbare Energien sowie der Energiebedarf nicht genau vorherzusagen sind [14].

2.4 Mathematische Grundlagen

Lineare Zusammenhänge werden in der Mathematik oft zur vereinfachten Betrachtung der Realität verwendet. In dieser finden sich prinzipiell auch Nichtlinearitäten, die nur sehr aufwendig modelliert und berechnet werden können. Erste Ansätze, um dies für die Optimierung von ES zu ermöglichen, existieren bereits. Der Großteil der entwickelten Softwares arbeitet dennoch mit den Ansätzen des linearen (LP)- bzw. gemischt-ganzzahligen linearen (MILP) Programmierens. So entsteht ein Kompromiss zwischen Berechnungsdauer der Optimierung und Detailgrad der Ergebnisse.

Wie bereits erläutert, werden modellierte ES in Gleichungssysteme überführt und zum Lösen an *Solver* übermittelt. Beispiele für frei zugängliche *Solver* sind der *CBC (Coin-or branch and cut) Solver* [15] und das *GLPK (GNU Linear Programming Kit)* [16]. Der *CBC-Solver* ist nach Bauer et al. [17] nur für weniger komplexe Energiesysteme verwendbar, da bei größeren Datenmengen die Berechnungszeit ansteigt. Steigt die Komplexität des Systems, empfehlen sie den *GLPK-Solver*. Allgemein arbeiten *Solver* mit den mathematischen Grundlagen des LP und MILP, welche in den folgenden Kapiteln erläutert werden.

2.4.1 Lineare Programmierung (LP)

Die lineare Optimierung (engl. *linear programming*) beschreibt das Minimieren bzw. Maximieren eines Systems $A \in \mathbb{R}^{n,m}$ durch anpassen der Variablen $x = (1, \dots, n)^T$ bei gleichzeitigem einhalten von Nebenbedingungen bezüglich $b = (1, \dots, m)^T$. Die Variablen beschreiben die Parameter die verändert werden können, um ein anderes Ergebnis zu erzielen und werden auch als Entscheidungsvariablen oder Aktivitäten bezeichnet. Weiterhin implizieren die Ziele der Kosten- und Emissionsminimierung, die Nebenbedingung der Nichtnegativität für x , welche bei der Optimierung berücksichtigt wird.

Somit ergibt sich

$$\begin{aligned} \min c^T \cdot x \\ A \cdot x \leq b \\ x \geq 0 \end{aligned} \tag{2.1}$$

die Beschreibung für eine Optimierung mit dem Ziel der Minimierung. Das Ergebnis ist also die kleinstmögliche Form der linearen Zielfunktion $c^T x$, bei Einhalten der restriktiven Nebenbedingung $Ax \leq b$. Angewandt mit den weiteren Relationsoperatoren lässt sich das System verallgemeinern

$$\begin{aligned} \min/\max c^T x \\ Ax \begin{cases} \leq \\ = \\ \geq \end{cases} b \end{aligned} \tag{2.2}$$

und ermöglicht die präzise Beschreibung aller Aktivitäten der jeweiligen Komponenten des ES für unterschiedliche Zielsetzungen. Beschrieben werden können Gleichung (2.1) und (2.2) zusammengefasst in Normalform

$$\min\{c^T x: Ax = b, x \geq 0\} \tag{2.3}$$

mit der Nebenbedingung der Gleichheit. Für diese Schreibweise sind alle weiteren Operatoren, Zielsetzungen und Nebenbedingungen formulierbar und ermöglichen eine übersichtliche Darstellung.

Die Theorie der linearen Optimierung besagt, dass jedes Gleichungssystem einen zulässigen Bereich besitzt, in dem alle Nebenbedingungen eingehalten werden. Sobald Gleichungen und Ungleichungen in ein Koordinatensystem aufgetragen werden, entsteht wie in Abbildung 2-1 verdeutlicht, ein Lösungsraum bzw. zulässiger Bereich, aus dem die optimale Lösung ermittelt wird. Diese muss auf einer der Ecken des zulässigen Bereichs P liegen, da nur in diesen ein Minimum oder Maximum bezüglich der Aktivität x eingehalten ist. Wird die jeweilige Ecke erreicht, kann die Zielfunktion nicht weiter verringert werden, ohne dass der zulässige Bereich verlassen wird. Diese Position beschreibt dann das Minimum, welches mit Hilfe der jeweiligen Zusammensetzung der Aktivitäten erreicht wird.

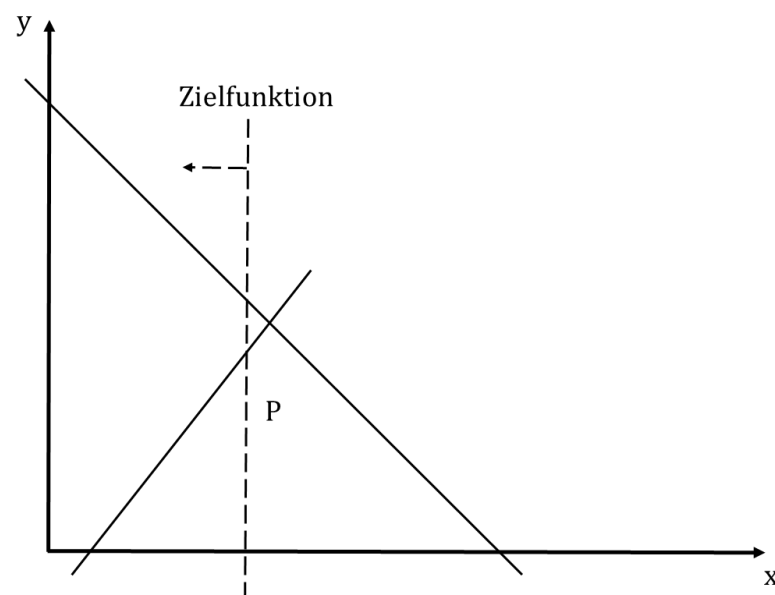


Abbildung 2-4: Zulässiger Bereich P einer linearen Optimierung zur Minimierung bezüglich der Aktivität x .

Sobald ein größeres ES betrachtet wird, besteht das Gleichungssystem aus einer höheren Anzahl von Komponenten, ihren Aktivitäten und jeweiligen Nebenbedingungen. Ein Gleichungssystem wird dann als Polyeder dargestellt, in dem die optimale Lösung auf einer der Ecken liegen muss. Ein Polyeder ist ein dreidimensionaler, vieleckiger Körper, der aus ebenen Flächen zusammengesetzt ist. Diese Flächen entstehen durch das Auftragen aller systembeschreibenden Gleichungen und Ungleichungen in ein Koordinatensystem [18].

Um Gleichungssysteme dieser Art lösen zu können, verwenden *Solver* das Simplex-Verfahren oder die Innere-Punkt Methode. Das Simplex-Verfahren ist ein Algorithmus, der die Zielwerte jeder Ecke des Polyeders Schritt für Schritt berechnet. Die Zielwerte werden

systematisch miteinander verglichen und der weniger optimale wird nicht weiter berücksichtigt. So bleibt zum Ende des Simplex-Verfahrens die bestmögliche Zielfunktion, um das Gleichungssystem im Hinblick auf die Optimierungsaufgabe zu lösen. In real beschriebenen Systemen können in der Regel viele Aktivitäten als 0 angenommen werden, wodurch diese mit Hilfe der Simplex-Methode effizient gelöst werden können. Sobald größere Systeme betrachtet werden, steigt auch die Komplexität. Nach Unger et al. [19] bedeutet Komplexität in dem Kontext des linearen Programmierens, dass eine erhöhte Rechenzeit benötigt wird, um den optimalen Zielwert zu ermitteln.

Bei ES kann, wie in Kapitel 2.1 beschrieben, davon ausgegangen werden, dass ein hohes Maß an Komponenten, Aktivitäten und ihren Nebenbedingungen berücksichtigt werden muss, um ein möglichst realistisches Modell zu erzeugen. Daher wird die Innere-Punkt Methode verwendet, um die Dauer zum Lösen der Gleichungssysteme zu verringern. Sie verfolgt einen ähnlichen Ansatz, wie die Simplex-Methode, mit dem Durchlaufen eines Polyeders. Allerdings erfolgt der Berechnungsweg nicht über die Ecken, sondern direkt durch das Innere des Polyeders. Um die Verfahren zu verdeutlichen, wird das Gleichungssystem aus Abbildung 2-4 mit zwei weiteren Gleichungen erweitert und in der folgenden Abbildung dargestellt.

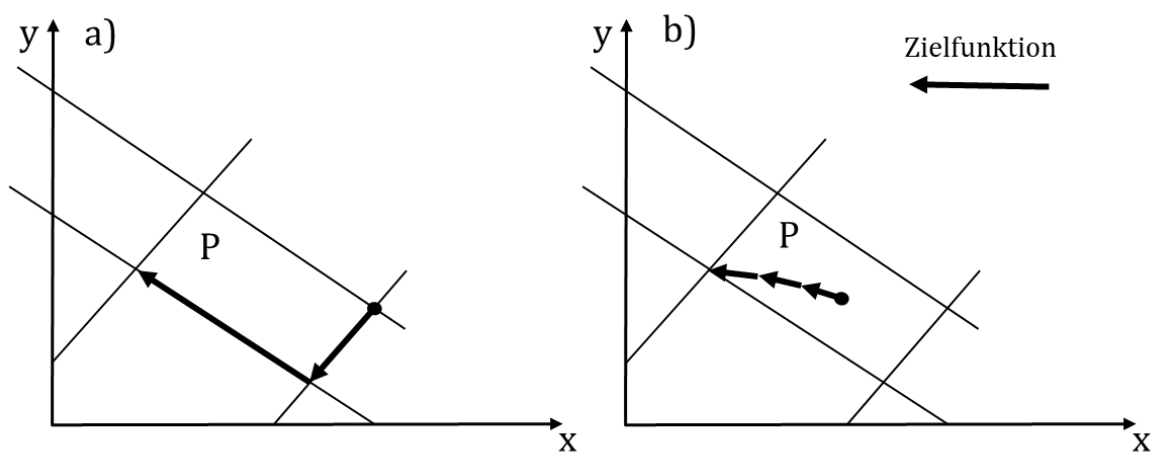


Abbildung 2-5: Schematische Darstellung a) des Simplex-Verfahrens und b) der Inneren-Punkt Methode für ein zu minimierendes System bezüglich der Aktivität x .

Für beide Verfahren gilt, dass ein Minimum bezüglich x gefunden werden soll. Bei der Simplex-Methode nähert sich der Zielfunktion der optimalen Lösung, indem die Kanten des zulässigen Bereichs P abgelaufen werden. Wohingegen die Innere-Punkt Methode in

diesem Bereich beginnt, ihn durchläuft und jeder weitere Punkt eine Kombination der Aktivitäten repräsentiert. Ist dieser besser hinsichtlich des Optimierungsziels, so wird er Grundlage für einen weiteren Vergleich. Durch den direkteren Weg kann die Effizienz gesteigert werden [19].

2.4.2 Gemischt-ganzzahlige lineare Programmierung (MILP)

Gemischt-ganzzahliges, lineares Optimieren (engl. *mixed-integer linear programming, MILP*) ist ein weiteres Teilgebiet für das Lösen linearer Gleichungssysteme und berücksichtigt ganzzahlige Entscheidungsvariablen. Da es in der Realität möglich ist, dass Variablen einer Komponente nicht teilbar sind, müssen diese diskreten Parameter berücksichtigt werden. Diskret bedeutet für diesen Fall, dass keine kontinuierliche Änderung der Parameter erfolgt, sondern ein fester Zustand über eine Periode von Zeitschritten angenommen wird [20]. Beispielsweise ermöglicht dies eindeutig zu definieren, ob eine Komponente in Betrieb ist oder ob eine Ressource zur Verfügung steht. Diese Ja/Nein-Entscheidung kann mit Hilfe der binären Schreibweise $y \in \{0, 1\}^m$ festgelegt und an das System übergeben werden. Ein System wie in Gleichung (2.1) würde demnach mit ganzzahligen Parametern erweitert werden. Es entsteht ein gemischt-ganzzahliges lineares Gleichungssystem,

$$f(x, y) \rightarrow \min \begin{cases} g(x, y) = 0 \\ h(x, y) \leq 0 \end{cases} \quad \text{für } x \in \mathbb{R}^n, y \in \{0, 1\}^m \quad (2.4)$$

mit der Zielfunktion f , den Gleichungen g und Ungleichungen h . Enthalten sind weiterhin die kontinuierlichen Aktivitäten x und ganzzahlige binäre Variable y . [17]

Möglichkeiten zum Lösen dieser linearen Gleichungssysteme sind die *branch-and-bound* Methode sowie das Schnittebenenverfahren. Im ersten Schritt wird bei beiden das Gleichungssystem unter Vernachlässigung der gemischt ganzzahligen Variablen gelöst. Dies erfolgt durch sogenannte LP-Relaxierung und bedeutet ein weniger striktes Einhalten der Grenzen, die während der linearen Programmierung erstellt wurden.

$$y \in \{0, 1\}^m \rightarrow 0 \leq y \leq 1 \quad (2.5)$$

Diese neue Bedingung wird auch als untere und obere Schranke bezeichnet. Durch diesen Schritt wird eine erste, meist nicht ganzzahlige Lösung ermittelt. Während bei dem

Schnittebenenverfahren eine zusätzliche Ungleichung verwendet wird, um den zulässigen Bereich zu verringern, wird bei der *branch-and-bound* Methode wie in Abbildung 2-6 das System (P_0) unterteilt (*branch*) und separat untersucht.

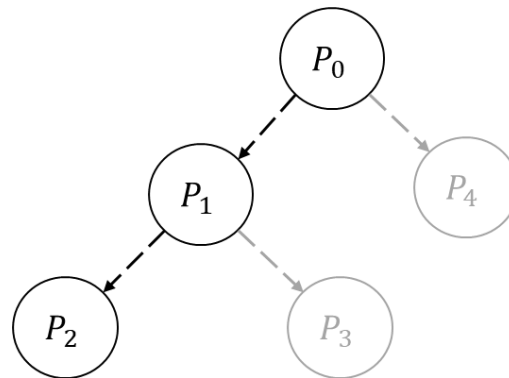


Abbildung 2-6: Zustandsbaum als Grundlage für die Branch-and-bound-Methode.

Ergibt die Untersuchung eines Teilsystems, dass dieses nicht lösbar ist (P_4) bzw. die Lösung nicht im zulässigen Bereich liegt (P_3), ist auch die optimale Lösung nicht in diesem Teilsystem enthalten und es wird nicht weiter berücksichtigt (*bound*). Weiteruntersucht wird ein lösbares System (P_1), indem dieses wieder durch Hinzufügen von Bedingungen, unterteilt wird. Kann das System nicht weiter unterteilt werden (P_2), so ist entweder die optimale ganzzahlige Lösung gefunden oder das gesamte System besitzt keine Lösung. Die Kombination aus Schnittebenenverfahren und *branch-and-bound* wird als *branch-and-cut* bezeichnet und steigert die Effizienz der Optimierung. [21]

2.5 Vergleich von Software zur Modellierung und Optimierung

Durch das erhöhte Angebot neu entwickelter Software zur Modellierung von ES ist die Notwendigkeit entstanden, diese zu differenzieren und einordnen zu können. Verglichen werden kann anhand äußerer Eigenschaften, den Charakteristika und anhand der erzielten Ergebnisse. Um die Effizienz eines Forschungsvorhabens zu steigern, ist es wichtig im Vorfeld zu erkennen, welche Software für eine spezielle Fragestellung geeignet ist. Durch unterschiedliche Zielsetzungen sowie variierender Parametrisierung entstehen Abweichungen zwischen den erzielten Optimierungsergebnissen. Diese sind jedoch nicht falsch, es existiert vielmehr ein Lösungsraum, in dem die ermittelten Ergebnisse liegen.

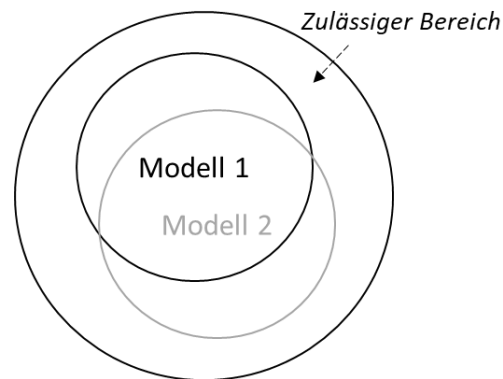


Abbildung 2-7: Überschneidung der Ergebnisse von Softwaremodellen im gleichen Lösungsraum.

Jedes Modell zeichnet sich durch eine eigene Formulierung von Technologien und dem jeweiligen Detailgrad aus. Es wird ein individueller Fokus gesetzt und je nach techno- ökonomischer Komplexität variiert Lage und Größe des Modells in dem zulässigen Bereich. Dennoch sollen in jedem Fall die Ergebnisse der Optimierung eine mögliche Realität so exakt wie möglich darstellen. Durch die Anwendung verschiedener Softwares und aufspannen des Lösungsraumes kann die Verlässlichkeit der Ergebnisse erhöht werden [11].

2.5.1 Stand der Forschung

Vergleichende Analysen von Software für die Optimierung von ES können anhand grundlegender Charakteristika sowie ihrer erzielten Ergebnisse durchgeführt werden. Connolly et al. [6] beschreiben den Vergleich von Software dieser Art als Notwendigkeit, um diese im richtigen Bereich anzuwenden. Sie vergleichen insgesamt 68 Modelle zur Untersuchung von ES initial anhand der Nutzeranzahl, in welchem Kontext die Software verwendet wird und ob diese frei zugänglich oder kommerziell ist. Tiefergehend betrachten sie Charakteristika, wie zum Beispiel, welche Sektoren untersucht werden können und ob ein verbrauchs- oder versorgungsorientierter Ansatz verfolgt wird. Sie differenzieren zwischen Modellen die entweder einen Simulations- oder einen Optimierungsansatz verfolgen. Zusätzliche eingrenzende Punkte sind die zeitliche und geographische Auflösung der Daten. Mittels ihrer Arbeit wollen sie einen Überblick ermöglichen, an denen sich Forscher und Entwickler orientieren können, um die passende Software für ihr Forschungsvorhaben auszuwählen [6].

Ringkjøb et al. [4] vergleichen 75 Softwaremodelle unter ähnlichen Aspekten, betrachten zusätzlich jedoch die jeweilige Programmiersprache, wie zum Beispiel Python oder Matlab. Außerdem beachten sie die Softwarearchitektur, wo sich Unterschiede hinsichtlich der Komponenten sowie ihrer sektoralen Zuordnung zeigen. Schließlich wird die mathematische Methode für die Lösung der modellierten Gleichungssysteme aufgeführt [4]. Der Einfluss des mathematischen Ansatzes wird von Ommen et al. [22] untersucht. Sie können zeigen, dass eine Steigerung der Genauigkeit von LP, über MILP zu NLP erzielt wird. Hinsichtlich der Rechenzeit ist jedoch MILP aktuell die geeignetste Lösungsmethode [22].

Verglichen werden können Softwaremodelle demnach auch hinsichtlich der erzielten Ergebnisse. Zum Beispiel zeigen Wilkerson et al. [23] mittels drei Modellen, dass die Verringerung von Emissionen zwischen Simulations- und Optimierungssoftware stark abweicht [23]. Priesmann et al. [24] vergleichen Modellergebnisse hinsichtlich der Komplexität des zuvor definierten ES. Eine komplexe Formulierung führt nach ihnen nicht unmittelbar zu genaueren Ergebnissen. Vielmehr nimmt mit zunehmender Komplexität die zusätzlich erzielte Genauigkeit ab [24]. Es muss folglich darauf geachtet werden, welche Parameter die Komponenten berücksichtigen können und welche nicht. Um Softwaremodelle geeignet vergleichen zu können, ist deshalb eine Harmonisierung der verwendeten Datengrundlage erforderlich. Das untersuchen Misconel et al. [25], indem sie drei Softwaremodelle mit Hilfe des gleichen Szenarios und harmonisierter Daten vergleichen. Sie zeigen, dass ähnliche Ergebnisse mit den Modellen erzielt werden und kommen zu der Schlussfolgerung, dass diese zur Analyse des ES geeignet sind. Dennoch arbeiten sie heraus, dass das Generieren einer harmonisierten Datengrundlage sowie der individuellen ES eine Herausforderung darstellt. Somit sollte nach ihnen ein Vergleich von Modellen eher mit weniger komplexen Energiesystemen und Szenarien geschehen [25]. Steinbrink et al. [26] arbeiten heraus, dass für komplexere ES eine automatisierte Erstellung für die unterschiedlichen Softwaremodelle notwendig ist. Weiterhin könne durch eine standardisierte Form der erzielten Ergebnisse zusätzlich die Plausibilitätsprüfung dieser vereinfacht werden. Auf Basis der Ergebnisse könnte dann eine Echtzeitüberwachung und Optimierung ermöglicht werden, indem geeignete Visualisierungen der Daten aus der Co-Simulation genutzt werden [26].

Diese Erkenntnis greifen Gils et al. [5] auf und vergleichen Ergebnisse unterschiedlicher Softwaremodelle anhand einer standardisierten Form, welche automatisiert erstellt wird. Sie stellen dabei fest, dass die gesamte umgesetzte Energiemenge über die untersuchte Zeitperiode einen ersten Überblick über die erzielten Modellergebnisse ermöglicht. Durch die Betrachtung von Erzeugungs- und Lastprofilen kann schließlich eine detailliertere Untersuchung erfolgen [5].

Angesichts des breiten Spektrums an Modellierungssoftware ist ein angemessener Vergleich erforderlich, um sicherzustellen, dass diese für die geeigneten Anwendungsgebiete eingesetzt wird. Bereits 2014 ist die Plattform *Open Energy Modelling Initiative (Open-Mod)* [27] entstanden, die frei zugängliche Software zur Modellierung von ES verzeichnet. Als Unterscheidungsmerkmale werden einige der bereits genannten grundlegenden Charakteristika aufgeführt. Dazu gehören der Modellierungsansatz, die sektorale Betrachtung sowie die zeitliche und örtliche Auflösung. Zusätzlich werden wichtige Daten und Veröffentlichungen der ES Forschung aufgeführt. Neue Modelle können von Entwicklern und Nutzern hinzugefügt werden, wodurch die gemeinschaftliche Arbeit erleichtert werden soll. Eine Anwendung der gewonnenen Erkenntnisse ist die Co-Simulation. Durch die Bündelung von Kompetenzen werden die Stärken individueller Softwares sinnvoll miteinander kombiniert. Das Framework *Mosaik* [28] (modulare Simulation aktiver Komponenten) ist speziell für komplexe ES und Smart Grids entwickelt [29].

Die erzielten Ergebnisse von Software zur Modellierung von ES können voneinander abweichen, weshalb es sowohl einen Bedarf als auch Möglichkeiten für einen Vergleich gibt. Bisherige Forschungsarbeiten zeigen, dass hierfür eine harmonisierte Datenbasis ebenso wichtig ist, wie die Übereinstimmung bestimmter Modelleigenschaften.

Als Quintessenz konnte somit herausgearbeitet werden, dass durch eine automatisierte und standardisierte Ergebnisdarstellung und Visualisierung der Nutzen eines Vergleichs weiter gesteigert werden könnte. Bisher gibt es kein Angebot für die Transformation von einheitlich formulierten ES zu der jeweiligen Software, um einen abschließenden Vergleich mit standardisierten Ergebnissen zu ermöglichen.

Das am Institut für Energietechnik der Technischen Universität Hamburg entwickelte Framework *Tessif* soll diese Ansätze aufgreifen und zum Teil Lösungsmöglichkeiten bereitstellen.

2.5.2 Transforming Energy Supply System (Modelling) Frameworks

Das Framework *Transforming Energy Supply System (Modelling) Frameworks (Tessif)* [30] ermöglicht es Energiesysteme für verschiedene Softwaremodelle zu transformieren und die Ergebnisse in einer standardisierten Weise zu präsentieren. Um die Übersetzung spezifischer ES und Darstellung zu ermöglichen, werden jeweils individuelle Funktionen¹ verwendet. In *Tessif* ist darüber hinaus eine Bibliothek von Beispielennergiesystemen enthalten, die speziell für den Vergleich von Software entwickelt worden sind. Weniger komplexe Beispiele werden außerdem bereitgestellt, um einen anfänglichen Eindruck von *Tessif* und der jeweiligen Software zu vermitteln. Soll ein spezifisches ES untersucht werden, bietet *Tessif* eine intuitive, interdisziplinäre Oberfläche für die Modellierung und Optimierung, um Entwicklern und Anwendern aus allen Bereichen einen schnellen Zugang zu ermöglichen. Gängige Praxis bei der Entwicklung von Energiesystemmodellen ist der modulare Aufbau aus Basiskomponenten (Kapitel 2.1), den auch *Tessif* nutzt. Die gesamte Transformation erfolgt, wie in Abbildung 2-8 dargestellt, mit der Funktion *es2es* (*energy-system to energy-system*), die für jede in *Tessif* integrierte Software individuell entwickelt wurde.

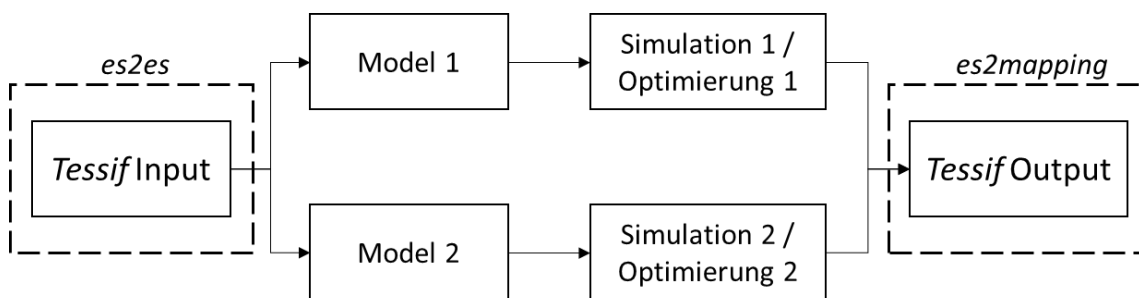


Abbildung 2-8: Schematische Darstellung der Einsatzweise von *Tessif* mit den Funktionen für die Übersetzung der Energiesysteme (*es2es*) und für die standardisierte Darstellung der Ergebnisse (*es2mapping*).

¹ Der Begriff Funktion bezieht sich in dieser Arbeit auf die Aufgabe, die durch *Tessif* erfüllt wird.

Die erzielten Ergebnisse können abschließend mit der Funktion *es2mapping* (*energy-system to mapping*) standardisiert dargestellt werden. Unterteilt werden können diese in Ergebnisse zum Gesamtsystem, zu den einzelnen Komponenten und zu den vorhandenen Energieflüssen. Bei der Untersuchung von komplexeren ES kann mit Hilfe einer Visualisierung (*generic_graph*) durch *Tessif* schnell und effizient eine erste Plausibilitätsprüfung der Ergebnisse durchgeführt werden. Beispielhaft ist im Folgenden das erweiterte ES aus Kapitel 2.1 Abbildung 2-3 dargestellt.

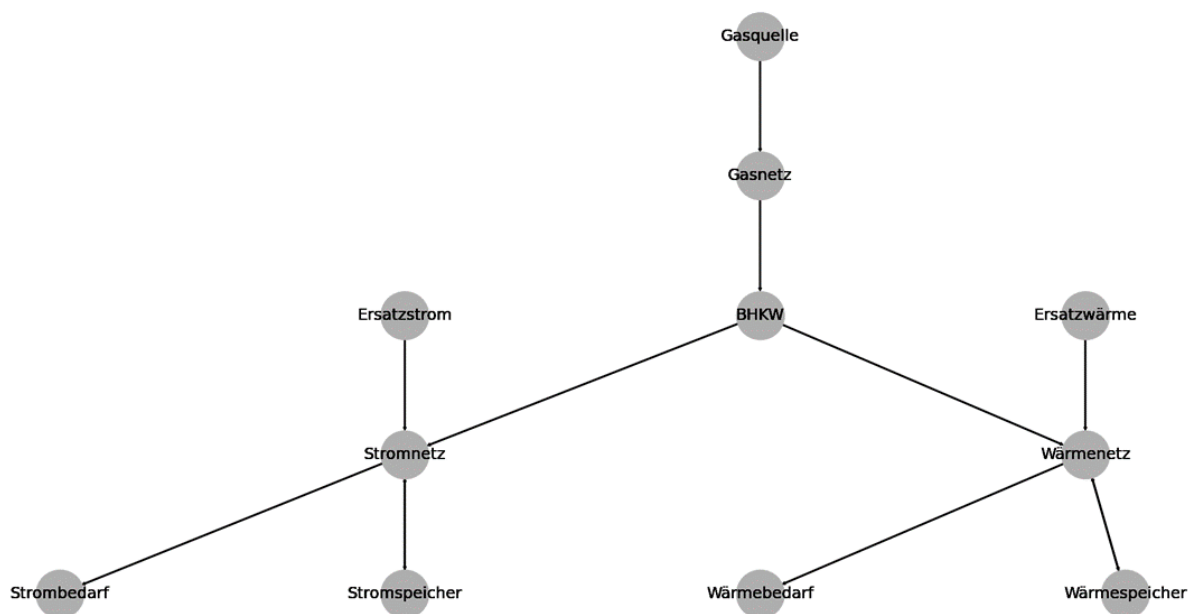


Abbildung 2-9: Graphische Darstellung des Energiesystems aus Abbildung 2-3 visualisiert mittels *Tessif*.

Eine Erweiterung (*advanced_graph*) dieser Darstellung kann durch die Skalierung der Komponenten und der Verbindungen erfolgen. Die Visualisierung des ES wird dann so erstellt, dass die Verhältnisse von Punkten und Linien die erzielten Ergebnisse repräsentieren. Die Graustufung der Verbindungen ist dann weiterhin ein Indikator für die jeweils auftretenden Emissionen bzw. Ressourcen. Bisher integrierte Softwaremodelle sind *Oemof* (*Open Energy System Modelling Framework*) [31]² und *PyPSA* (*Python for Power System Analysis*) [32]³, die beide in Python basierend auf dem *bottom-up* Ansatz zur Modellierung und Optimierung von Versorgungssystemen entwickelt werden. *Oemof* ist

² <https://oemof.readthedocs.io/en/latest/>

³ <https://pypsa.readthedocs.io/en/latest/index.html#>

eine gemeinschaftlich entwickelte Software für die Untersuchung ganzheitlicher ES und *PyPSA* weist speziell Stärken für den Stromsektor auf. Sie ermöglichen beide eine gekoppelte Betrachtung der Sektoren und unterstützen eine hohe zeitliche und geographische Auflösung der Daten. Für eine genauere Erläuterung der Softwaremodelle sei auf die jeweilige Dokumentation verwiesen.

Ziel von *Tessif* ist es, einen Rahmen zu schaffen, der von Ingenieuren und Informatikern gleichermaßen genutzt werden kann, um Softwaremodelle und ES zu vergleichen und die Ergebnisse auszuwerten. Dafür sollen ES automatisiert in die jeweilige Software transformiert werden, durch welche die Optimierung dann erfolgt. Anschließend werden durch *Tessif* die Ergebnisse gezielt entnommen und in einer standardisierten Form dargestellt.

2.6 Framework for Integrated Energy System Assessment

Das *Framework for Integrated Energy System Assessment (FINE)* [33]⁴ ist eine gemischt-ganzzahlige, lineare Optimierungssoftware zur Analyse von Energiesystemen. *FINE* wurde vom Forschungszentrum Jülich entwickelt und im Rahmen des Projekts *Energie System 2050 (ES2050)* [34] der *Helmholtz Energy Computing Initiative (HECI)* unterstützt. Die Initiative listet *FINE* und weitere frei zugängliche Software auf einer Plattform, die von den Helmholtz-Zentren und ihren Partnern für die Untersuchung von Energiesystemen entwickelt und eingesetzt wird. Zu den Empfehlungen gehören darüber hinaus beispielsweise das Softwaremodell *PyPSA* sowie das Zeitreihenaggregationsmodul *Tsam*. Zusätzlich führen sie, ähnlich wie *OpenMod*, Software auf, die für den Zugriff auf Erzeugungs- und Verbrauchsdaten verwendet werden kann [35].

FINE verfolgt den Ansatz, die Kosten eines modular erstellten Energiesystems zu minimieren und gleichzeitig festgelegte ökologische oder politische Restriktionen einzuhalten. Durch die Basiskomponenten Quelle (*Source*), Senke (*Sink*), Speicher (*Storage*), Energieumwandlung (*Conversion*) und Energietransport (*Transmission*) können individuelle

⁴ <https://vsa-fine.readthedocs.io/en/latest/index.html>

Systeme für eine Vielzahl von Forschungsthemen erstellt werden. Mit Hilfe einer Optimierung wird dann für jedes Objekt die bestmögliche Dimensionierung und Auslastung berechnet. [36]

2.6.1 Komponenten

Ein Energiesystem in *FINE* kann aus den in Kapitel 2.1 bereits eingeführten fünf Basis-komponenten modular zusammengestellt werden. Es entsteht ein Bilanzraum mit ein- und austretenden Energieflüssen, interner Energieumwandlung und -speicherung, der je nach Zusammenstellung einfache bis komplexe Themen beschreiben kann. Im Folgenden wird jede Komponente kurz erläutert. Für eine detaillierte Beschreibung sei auf die Arbeit von Welder et al. verwiesen [36].

2.6.1.1 Energietransport (*Transmission*)

Der Energietransport beschreibt die Verbindung zwischen zwei Komponenten oder Regionen, um einen bestimmten Energieträger zu übermitteln. In *FINE* ist es möglich Verluste pro Längeneinheit als technischen Parameter festzulegen und zusätzlich die Leistung zu begrenzen. So können reale Transportwege möglichst detailliert beschrieben werden.

2.6.1.2 Quellen und Senken (*Sources and Sinks*)

Quellen werden als Ausgangspunkt von Energieflüssen in einem Energiesystem verwendet. Diese liefern Energie in die Bilanzgrenze und damit zu den Komponenten, die die jeweilige Energiequelle nutzen, umwandeln oder speichern. Eine Quelle kann durch die Hinterlegung von Erzeugungsprofilen, skaliert auf die installierte Leistung, näher beschrieben oder begrenzt werden. Senken hingegen sind der Endpunkt eines Energieflusses und führen den jeweiligen Energieträger aus dem Bilanzraum heraus. Sie beschreiben den Bedarf, der durch die weiteren Komponenten gedeckt werden muss.

2.6.1.3 Energieumwandlung (*Conversion*)

Durch eine Energieumwandlung kann ein bereits im System vorhandener Energieträger umgewandelt werden, in einen oder mehrere weitere. Diese Prozesse sind in der Realität verlustbehaftet, weshalb es erforderlich ist, einen Wirkungsgrad für die Komponente zu

definieren. Durch die Umwandlung von fossilen Energieträgern müssen weiterhin spezifische Emissionen zugeordnet werden.

2.6.1.4 Speicher (*Storage*)

Speichersysteme werden eingesetzt, um überschüssige Energie temporär zu lagern und zu einem späteren Zeitpunkt wieder zur Verfügung zu stellen. Dies ist immer für genau einen Energieträger möglich, der in der Regel verlustbehaftet ein- und ausgespeichert wird. Während der Speicherdauer muss außerdem eine mögliche Selbstentladung berücksichtigt werden, um eine saisonale Speicherung darzustellen.

2.6.2 Fähigkeiten und Grenzen

FINE zeichnet sich insbesondere durch eine sehr gute Zugänglichkeit und Semantik aus. Sowohl die einzelnen Komponenten mitsamt Parametern als auch das Energiesystemmodell selber sind klar formuliert, sodass die Einarbeitung wenig zeitaufwendig ist. Des Weiteren sind diverse Beispiele vorhanden, die verschiedene Maßstäbe von Systemen untersuchen. Auch die Kopplung der Sektoren Strom, Wärme und Verkehr wird dargestellt und insbesondere in Kombination mit Wasserstoff beschrieben. Die Flexibilität in der Formulierung der Komponenten ermöglicht darüber hinaus eine Anwendung in weiteren Forschungsthemen, wie zum Beispiel im Bereich von Verteilungsnetzen. Somit bietet *FINE* eine große Vielfalt an Möglichkeiten zur Untersuchung der Transformationspfade des gesamten Energiesystems und weiterer Infrastrukturen. Temporal können alle Zeitformate als Datengrundlage dienen. Sie müssen lediglich für alle weiteren Daten einheitlich übergeben werden. Geographisch ist es möglich, einzelne miteinander kommunizierende Systeme zu beschreiben oder auch ein einzelnes Gesamtsystem. Der Detailgrad der Komponenten weist sich durch die Vielfalt der ökonomischen Parameter aus. Insbesondere die berücksichtigte Verteilung der Investitionskosten auf die Lebensdauer sowie der jeweilige Zinssatz zeichnet *FINE* aus. Werden die technischen Parameter betrachtet zeigen sich umfangreiche Möglichkeiten zur Beschreibung von Komponenten. Neben den minimal installierten Leistungen bzw. Kapazitäten können Erzeugungs- und Lastprofile vorgegeben werden, um ein spezifisches Verhalten zu definieren. Komponenten können auch voneinander abhängig gemacht werden, um beispielsweise Verbraucher und Speicher zu ver-

binden. Werden Energieumwandlungen betrachtet, ist es möglich, diese in *FINE* limitierend zu modellieren, indem ein maximaler Lastwechsel pro Zeitschritt definiert wird. Allerdings ist es nach aktuellem Wissensstand bisher nicht möglich, einem Lastwechsel spezifische Kosten beizumessen. Des Weiteren zeichnen sich reale Kraftwerksprozesse durch nichtlineares Verhalten aus, weshalb Teillastverhalten durch eine abschnittsweise Linearisierung in *FINE* berücksichtigt werden kann.

2.6.3 Time Series Aggregation Module (*tsam*)

Sehr komplexe Energiesysteme sind häufig mit einer erhöhten Berechnungszeit für eine Optimierung verbunden. Um einen Kompromiss zwischen Zeitaufwand und Detailgrad der Ergebnisse zu ermöglichen, kann entweder die Beschreibung der Komponenten oder die Datenbasis der Zeitreihen simplifiziert werden. In beiden Fällen geschieht dies mittels einer Gruppierung, auch bezeichnet als Aggregation, von Informationen die Ähnlichkeit aufweisen. Multiple Erzeuger- oder Verbraucher können demnach zu einer übergeordneten Komponente zusammengefasst werden, sofern Übereinstimmungen in der technischen Beschreibung vorhanden sind. Als Beispiel sei das Zusammenfassen von Windkraftanlagen zu einem virtuellen Kraftwerk oder von diversen Verbrauchern des gleichen Energieträgers zu einem einzelnen Großverbraucher genannt. Die Aggregation von Zeitreihen hingegen beschreibt das systematische Zusammenführen von redundanten Informationen innerhalb der jeweiligen untersuchten Zeitspanne.

Das *time series aggregation module (tsam)* [37]⁵ ist ein Python-Package und gruppiert ebendiese gleichartigen Informationen mit Hilfe von Algorithmen, wodurch die temporale Komplexität verringert werden kann. Es existieren unterschiedliche Vorgehensweisen, wie die Aggregation erfolgt. Zum einen kann direkt die Anzahl der Zeitschritte verringert werden, indem beispielsweise jeder zweite Zeitschritt betrachtet wird (*Downsampling*) oder es wird eine repräsentative Periode genutzt, um zuvor eingeteilte Zeitabschnitte zu beschreiben (*Clustering*). Aus einem gruppierten Abschnitt wird ein repräsentativer

⁵ <https://tsam.readthedocs.io/en/latest/>

Punkt bzw. Zeitabschnitt mit ähnlichen Informationen zusammengefasst, sodass schließlich eine verringerte Anzahl an Zeitschritten untersucht werden muss [38].

Die Aggregation der Zeitreihen von Energiesystemen basiert darauf, dass Erzeugungs- und Verbrauchsprofile oft aus einem sich wiederholenden Rhythmus bestehen. Eingeteilt werden im Vorfeld Zeitabschnitte, die besondere Ähnlichkeit zueinander aufweisen, auch bezeichnet als Cluster. Jedes Cluster wird einzeln untersucht und ein möglichst repräsentativer Mittelwert gewählt, um diesen gruppierten Zeitabschnitt mit möglichst geringem Informationsverlust zu beschreiben. Aufgrund der Periodizität im Energiesektor werden in der Regel Typtage als repräsentative Zeitabschnitte für die Aggregation verwendet. Da ausschließlich temporale Informationen aggregiert werden, kann gleichzeitig sichergestellt werden, dass keine relevanten technischen Informationen verloren gehen [39].

FINE ermöglicht eine direkte und bereits integrierte Anwendung von *tsam*, um alle Zeitreihen des übergebenen Energiesystems zu aggregieren und so die Optimierung zu beschleunigen. Standardmäßig wird aufgrund häufiger Periodizität im Energiesektor, für ein Jahr mit 8760 Stunden über jeweils 7 Tage mit 24 Stunden aggregiert. Berücksichtigt werden muss jedoch, dass durch die Mittelwertbildung vorhandene Extrempunkte vernachlässigt und somit Spitzenlasten unterschätzt werden. Deshalb kann die Aggregation zu einer Abweichung der Gesamtkosten führen, welche in der Regel zu niedrig ausfallen. Darüber hinaus erscheint die Abbildung von saisonal modellierten Speichern durch die Einteilung in Perioden kompliziert und wird momentan noch weiterentwickelt. Für eine umfassendere Beschreibung und Erläuterung weiterer Möglichkeiten für die Aggregation mittels *tsam* sei auf die Ausarbeitungen von Kotzur et. al. verwiesen [38].

Da in Zukunft ein Kompromiss zwischen zeitlichem Aufwand und dem Detailgrad der Ergebnisse gefunden werden muss, ist es sinnvoll, Werkzeuge für die Vereinfachung zu verwenden, welche dennoch die erzielten Ergebnisse der Optimierung nur in geringem Maße beeinflussen sollten.

3 Methodik

Das folgende Kapitel soll den Prozess der Auswahl und Integration eines neuen Softwaremodells in *Tessif* erläutern. Nachdem eine passende Software gefunden wurde, muss eine Funktion entwickelt werden, um ein Energiesystem (ES) aus dem Quellformat *Tessif* in das Zielformat der Software zu transformieren. Vereinfacht beschrieben werden kann der Vorgang als eine Übersetzung von *Tessif* zu der jeweiligen Software. Hierfür wurde jede Komponente detailliert untersucht, um relevanten Parameter zu identifizieren. Die Aufgaben von *Tessif*, bezeichnet als Funktionen, werden in der folgenden Abbildung 3-1 schematisch dargestellt.

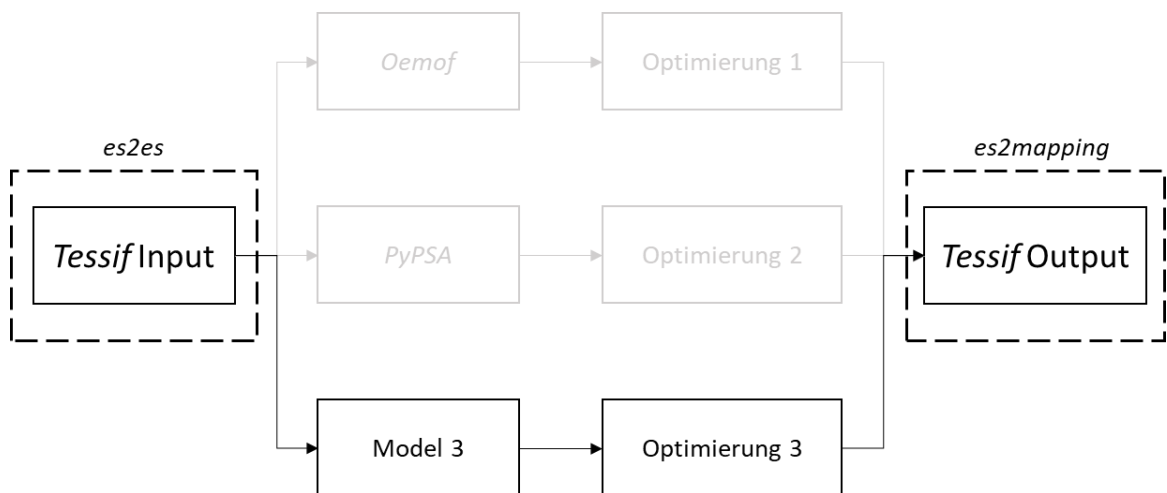


Abbildung 3-1: Schematische Darstellung der Einsatzweise von *Tessif*, erweitert mit dem Ziel der Arbeit.

Die Übersetzung erfolgt mittels der Funktion *energy-system to energy-system (es2es)*, welche den ersten durch *Tessif* ausgeführten Arbeitsschritt durchführt. Die zweite in dieser Arbeit entwickelte Funktion dient der standardisierten Darstellung der Optimierungsergebnisse. Diese sollen als Grundlage dienen, um Softwaremodelle miteinander vergleichen zu können. Als *Mapping* wird das strukturierte Ablegen von relevanten Daten bezeichnet, weshalb die Funktion als *energy-system to mapping (es2mapping)* betitelt wird.

Das Vorgehen bei der Integration wird in den weiteren Kapiteln detailliert beschrieben, um zu verstehen, wie eine Transformation durch *Tessif* ermöglicht wurde. Um eine Software zu integrieren, muss im Vorfeld eine geeignete Auswahl getroffen werden.

3.1 Auswahl des Softwaremodells

Um auswählen zu können, welche Software in *Tessif* integriert werden soll, wurden zunächst notwendige allgemeine Faktoren ermittelt. Das für diese Arbeit gewünschte Softwaremodell sollte Python als Programmiersprache aufweisen, da sowohl *Tessif* als auch die bisher verwendeten Modelle Python-basiert sind. Zur Unterstützung des Optimierungsprozesses werden weiterhin Python-Pakete (*packages*) verwendet. *Pyomo* [40] wird verwendet, um ES in Gleichungssysteme zu überführen, die durch die *Solver* optimiert werden können. Je nach Paketversion können die Softwaremodelle mit ihnen arbeiten oder es sind Anpassungen erforderlich. Um die zu integrierende Software möglichst in ihrer ursprünglichen Form nutzen zu können, wurde festgelegt, dass die Paketversionen kompatibel sind. Zusätzlich wird als äußerer Faktor berücksichtigt, dass das neue Softwaremodell mit Hilfe von in *Tessif* verwendeten, frei zugänglichen *Solvern* arbeitet. Die beiden bisher in *Tessif* integrierten Softwares verwenden die MILP-Methode (vgl. Kapitel 2.4.2) zum Lösen der Gleichungssysteme. Weitere einzuhaltende Charakteristika sind nach Kapitel 2.1 der *bottom-up* Ansatz, die Möglichkeit der Szenarioanalyse sowie das Untersuchen des transformierten ES mit gekoppelten Sektoren. Zeitlich und räumlich sollte das neue Softwaremodell die gleiche Auflösung erlauben, wie sie in *Tessif* formuliert werden kann. Die zur Auswahl eines geeigneten Softwaremodells festgelegten äußeren Faktoren und Charakteristika sind in der folgenden Tabelle 3-1 übersichtlich für die vorhandenen und das neu in *Tessif* zu integrierende Softwaremodell zusammengestellt.

Tabelle 3-1: Übersicht der Auswahlkriterien für in *Tessif* zu integrierende Software.

Kriterien	<i>Oemof</i>	<i>PyPSA</i>	<i>FINE</i>
Python basiert	+	+	+
Package-Versionen	+	+	+
Solver	+	+	+
LP/MILP	+	+	+
<i>Bottom-up</i> Ansatz	+	+	+
Szenarioanalyse	+	+	+
Sektorenkopplung	+	+	+
Optimierungsziel	+	+	+
Zeitliche Auflösung	+	+	+
Aktualität	+	+	+

Das *Framework for Integrated Energy System Assessment (FINE)* (vgl. Kapitel 2.6) erfüllt alle Kriterien und ist daher eine adäquate Software, um in *Tessif* integriert zu werden. Alle betrachteten äußeren Faktoren und Charakteristika stimmen überein und darüber hinaus wird *FINE* aktuell als Teil einer Initiative für frei zugänglicher Software weiterentwickelt. Insbesondere zeichnet sich *FINE* durch eine gute Zugänglichkeit der Softwaremodellarchitektur aus. Somit wird *FINE* ausgewählt, um in *Tessif* integriert und anschließend mit den weiteren Softwaremodellen verglichen zu werden.

3.2 Integration des Softwaremodells

Um die ausgewählte Software über *Tessif* nutzen zu können, müssen zwei Funktionen entwickelt werden, um sowohl die Übersetzung des ES durchzuführen als auch die Ergebnisse standardisiert darstellen zu können. Grundlage der Standardisierung sind die in Tabelle 3-2 aufgeführten *Unique Identifier (UID)*, die es ermöglichen, jede Komponente eindeutig zu beschreiben.

Tabelle 3-2: *Unique Identifier (UID)* für die eindeutige Beschreibung einer Komponente des Energiesystems.

<i>UID</i>	<i>Tessif</i>	Beispiel
Name	<i>name</i>	Windturbine
Energieträger	<i>carrier</i>	Elektrizität
Sektor	<i>sector</i>	Strom
Komponente	<i>component</i>	Quelle
Region	<i>region</i>	Hamburg
Breitengrad	<i>latitude</i>	53.5101
Längengrad	<i>longitude</i>	10.0312

Neben dem Namen und dem Energieträger der jeweiligen Komponente können auch Informationen über die sektorale Zugehörigkeit und den geographischen Standort angegeben werden. Die eindeutige Beschreibung der Komponenten vermeidet Mehrdeutigkeiten und vereinfacht die automatisierte Übersetzung und die einheitliche Darstellung der Ergebnisse.

3.2.1 Energy-System to Energy-System (*es2es*)

Um ein ES mit *Tessif* transformieren zu können, muss zuerst ein *FINE*-spezifisches ES erzeugt werden, das dann sukzessive mit allen Komponenten gefüllt wird. Initial müssen Informationen zu vorhandenen Energieträgern, den jeweiligen Einheiten sowie Informationen zur geographischen und zeitlichen Auflösung beschrieben werden. Weil mit *Tessif* erzeugte Komponenten in einer Gesamtregion modelliert werden, wird eine abstrakte Region festgelegt und zu den Standardeinstellungen hinzugefügt. Die in der folgenden Tabelle 3-3 aufgeführten Standardeinstellungen werden für das *FINE* ES übernommen und können durch den Benutzer angepasst werden.

Tabelle 3-3: Energiesystemparameter durch die Übersetzungsfunktion (*es2es*) in *Tessif* erstellt.
*Standardeinstellung von *Tessif*.

<i>ES Parameter</i>	Inhalt
<i>Locations</i>	Default_Region*
<i>commodities</i>	Alle Energieträger des Energiesystems
<i>commodityUnitsDict</i>	Einheiten der Energieträger
<i>numberOfTimesteps</i>	Anzahl der untersuchten Zeitschritte
<i>hoursPerTimeStep</i>	1*
<i>CostUnit</i>	€*

Um die Informationen zu den Energieträgern und ihren Einheiten zu erhalten, werden die *UID*'s aller in dem *Tessif* ES enthaltenen Komponenten durchsucht. Die Einheit kann erneut den Standardeinstellungen von *Tessif* entnommen und in Kombination mit dem Energieträger abgespeichert werden. Sollten Emissionen oder Ressourcen begrenzt sein, so müssen sie sowohl in den Energieträgern als auch in den Einheiten des ES zu finden sein. Die Funktion ist so formuliert worden, dass jedes ES mit diesen zur Optimierung notwendigen und individuellen Daten korrekt erzeugt wird.

Anschließend muss zugeordnet werden, welche Komponente aus *Tessif* die passende Komponente im *FINE* ES darstellt. Nach Kapitel 2.1 werden zur Modellierung von ES fünf Basiskomponenten verwendet, die sowohl in *Tessif* als auch in *FINE* zu finden sind. Sie zeichnen sich durch das gleiche Rahmenwerk aus, welches ihnen ermöglicht mit den je-

weiligen Energieträgern umzugehen. *Tessif* stellt zusätzlich zu den Basiskomponenten sogenannte Konnektoren zur Verfügung, welche als Verbindung zwischen zwei Potential-ebenen betrachtet werden können. Die folgende Tabelle 3-4 stellt alle Komponenten übersichtlich zusammen und führt gleichzeitig die *Tessif*- und *FINE*-spezifische Benennung auf.

Tabelle 3-4: Bezeichnung der Komponenten in *Tessif* und *FINE*. *Indirekt formulierbar

Komponente	<i>Tessif</i>	<i>FINE</i>	Beschreibung
Energieübertragung	<i>Bus</i>	<i>Transmission</i>	Verbindung von Komponenten
Quelle/Senke	<i>Source/Sink</i>	<i>Source/Sink</i>	Erzeugung und Bedarf
Umwandlung	<i>Transformer</i>	<i>Conversion</i>	Umwandeln einer Energieform
Speicher	<i>Storage</i>	<i>Storage</i>	Speicherung eines Energieträgers
Konnektor	<i>Connector</i>	<i>Conversion*</i>	Verbindung von Potentialebenen

Mit dieser Grundlage wird über die *es2es*-Funktion jede *Tessif* Komponente übersetzt und je nach Zugehörigkeit behandelt. Dafür wird die techno-ökonomische Parametrisierung jeder Komponente mit den jeweiligen Bestimmungen verglichen, die diese für *Tessif* oder *FINE* erfüllen. Parameter können sich in der Namensgebung der Modellsemantik unterscheiden, verfolgen jedoch häufig die gleiche Intention. Nacheinander wird daher jede Komponente analysiert und aus *Tessif* zu *FINE* transformiert. Das Prinzip wird in Abbildung 3-2 detaillierter erläutert.

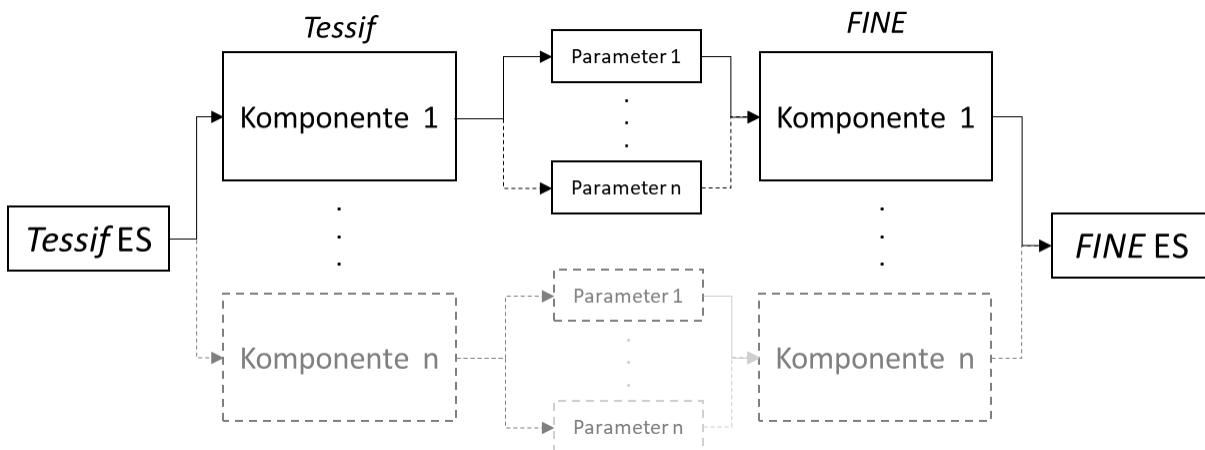


Abbildung 3-2: Schematische Darstellung der Transformation der Komponenten und zugehörigen Parametern von *Tessif* zu *FINE*.

Vorab muss für jede *FINE* Komponente festgelegt werden, ob ihre Leistung bzw. Kapazität durch die Optimierung angepasst werden darf. Es kann davon ausgegangen werden, dass dies im Sinne des übergeordneten Ziels der Kostenminimierung ermöglicht wird. Lediglich für Verbraucher und bei fest vorgeschriebenen Erzeugungsprofilen wird die Variation nicht erlaubt. Der jeweilige Variabilitätsparameter (*hasCapacityVariable*) wird durch die *es2es*-Funktion bestimmt und in der jeweiligen Komponente abgelegt. In *FINE* können daher sowohl Einsatzplanoptimierungen als auch Ausbauszenarien entwickelt werden. Zusätzlich wird in *FINE* über der Annuitätenmethode standardmäßig mit einem Zinssatz (*interestRate*) von 0,08 % und einer ökonomischen Lebensdauer (*economicLifetime*) von 10 Jahren gerechnet. *Tessif* und die bisher integrierten Softwaremodelle erfordern eine manuelle Bereitstellung der Annuitäten durch die jeweiligen Parameter, um vergleichbare Ergebnisse zu ermitteln. Deshalb wird der Zinssatz für die Integration von *FINE* auf 0 % gesetzt und die ökonomische Lebensdauer auf die Länge der untersuchten Zeitspanne begrenzt. Die drei bisher definierten Parameter werden in den folgenden Tabellen nicht aufgeführt, um Übersichtlichkeit zu gewährleisten.

Weiterhin müssen über die *es2es*-Funktion *UID*'s, spezifische Emissionen und Kosten im *FINE* ES abgespeichert werden, um diese später bei der Ergebnisdarstellung nutzen zu können. In *Tessif* können einzelne Kosten und Emissionen den spezifischen Energieflüssen einer Umwandlung zugeordnet werden. *FINE* hingegen bezieht sich während der Parametrisierung ausschließlich auf den eintretenden Fluss. Durch die Übersetzung werden daher alle Werte aus *Tessif* mit Hilfe der jeweiligen Wirkungsgrade zusammengefasst für einen übergeordneten Wert in *FINE*. Nachträglich ist es nicht möglich festzustellen, wie diese Werte berechnet worden sind. Deshalb wird das System lokal um die entsprechenden Informationen zu spezifischen Kosten und Emissionen erweitert, anhand derer später rekonstruiert werden kann, wie die Parameter entstanden sind. Der Prozess der Optimierung wird durch diese Erweiterung nicht beeinflusst.

Aufgeteilt wird die Integration von *FINE* in Basiskomponenten und indirekte Komponenten, die jeweils durch ihre spezifischen Parameter technisch und wirtschaftlich definiert werden. *Tessif* ermöglicht es jeder Komponente spezifische Emission zuzuordnen, wohin-

gegen dies in *FINE* ausschließlich für Energieumwandlungen vorgesehen ist. Um die Realität möglichst exakt darzustellen, sind daher indirekte Komponenten notwendig, die durch eine Kombination von Basiskomponenten einen erweiterten Nutzen erzielen.

3.2.1.1 Integration der Basiskomponenten

Die erste zu übersetzende Komponente ist der *Bus* in *Tessif* als Energietransport, zu der *Transmission* in *FINE* (vgl. Tabelle 3-4). Die zu übersetzenden Parameter dieser zwingend erforderlichen Komponente können Tabelle 3-5 entnommen werden.

Tabelle 3-5: Spezifische Parameter des Energietransportes für *Tessif* (*Bus*) und *FINE* (*Transmission*).

<i>Tessif</i>	<i>FINE</i>	Beschreibung
<i>name</i>	<i>name</i>	Benennung
<i>carrier</i>	<i>commodity</i>	Energieträger

Für *Tessif* ist in der *Bus*-Komponente abgelegt, welche Komponenten welchen Energieträger einspeisen bzw. entnehmen. Für *FINE* ist die Herkunft nicht relevant, es ist ausreichend den Energieträger (*commodity*) zu definieren. Da der Energietransport von *Tessif* als ideal angesehen werden, haben sie keinen Einfluss auf den Energieträger, sondern übertragen diesen ausschließlich zwischen Komponenten, die diesen gleichen Energieträger verwenden. Der jeweilige Energietransport steht bei der Optimierung uneingeschränkt zur Verfügung und die Transformation ist nach der Definition von Name und Energieträger für den Energietransport abgeschlossen.

Die nächste zu integrierende Komponente wird in beiden Softwares als Senke (*Sink*) bezeichnet und repräsentiert die Verbraucher. Alle zugehörigen Parameter zur Beschreibung befinden sich in der folgenden Tabelle 3-6.

Tabelle 3-6: Spezifische Parameter der Senke für *Tessif* (*Sink*) und *FINE* (*Sink*).

<i>Tessif</i>	<i>FINE</i>	Beschreibung
<i>name</i>	<i>name</i>	Benennung
<i>carrier</i>	<i>commodity</i>	Energieträger
<i>timeseries</i>	<i>operationRateFix</i>	Verbrauchsprofil

Die Integration von *FINE* arbeitet aktuell mit fest vorgeschriebenen Verbrauchsprofilen, die als *operationRateFix* übermittelt werden. Ist kein Verbrauchsprofil von *Tessif* vorgegeben, ermittelt die Übersetzung einen festen Verbrauch pro Zeitschritt auf Basis des maximalen Energieflusses (*flow_rate*) von *Tessif*.

Quellen sind das Pendant zu Senken, indem sie dem Bilanzraum Energie hinzufügen. Sofern eine Limitierung vorgegeben ist, müssen für Quellen leistungsbegrenzende Parameter, wie in Tabelle 3-7 übersetzt werden.

Tabelle 3-7: Spezifische Parameter der Quelle für *Tessif* (Source) und *FINE* (Source).

<i>Tessif</i>	<i>FINE</i>	Beschreibung
<i>name</i>	<i>name</i>	Benennung
<i>carrier</i>	<i>commodity</i>	Energieträger
<i>timeseries</i>	<i>operationRateMax</i>	Erzeugungsprofil
<i>flow_rates.min</i> <i>expansion_limit.min</i>	<i>capacityMin</i>	Minimale Leistung
<i>flow_rates.max</i> <i>expansion_limit.max</i>	<i>capacityMax</i>	Maximal Leistung
<i>flow_cost</i>	<i>opexPerOperation</i>	Betriebskosten (Opex)
<i>expansion_cost</i>	<i>InvestPerCapacity</i>	Investitionskosten (Capex)

Diese können aus der jeweiligen *Tessif* Komponente anhand der Parameter des Energieflusses (*flow_rates*), der ausbaubaren Leistung (*expansion_limit*) oder des Erzeugungsprofils (*timeseries*) abgeleitet werden. Die *es2es*-Funktion ist so formuliert, dass der passende Wert der minimalen und maximalen Leistungsbegrenzung automatisch ausgewählt wird. Ist ein Ausbau der Komponente bzw. Leistung möglich, so wird als maximal installierbare Leistung (*capacityMax*) der maximale Ausbau festgelegt. Sollte keine Leistung ausgebaut werden dürfen, orientiert sich die maximale Leistung an dem möglichen Energiefluss der Komponente.

Ist ein Erzeugungsprofil vorgegeben, wird pro Zeitschritt eine auf die Leistung normierte maximale Betriebsrate (*operationRateMax*) für den Zeitrahmen abgelegt. Für einen unbegrenzten Ausbau wird kein maximaler Wert festgelegt und das Maximum wird während der Optimierung ermittelt. Gemäß dem Fall, dass weder durch den Energiefluss noch durch den erlaubten Ausbau ein maximaler Wert vorgegeben ist und ein Erzeugungsprofil vorhanden ist, wird das Profil auf der Basis des eigenen Maximums normiert. Die minimal

erforderliche Leistung wird analog mit den minimalen Werten aus *Tessif* festgelegt. Das gleiche Vorgehen für die Bestimmung der Leistungsparameter wird bei allen weiteren Komponenten integriert. Die letzten zu übersetzenden Parameter der Quelle sind ökonomisch und können ohne Anpassungen übersetzt werden.

Als Nächstes wird der *Transformer* in *Tessif*, zu der *Conversion* in *FINE* übersetzt. Diese Komponente, mit zugehörigen Parametern aus Tabelle 3-8, beschreibt einen typischen Energieumwandlungsprozess von einer Energieform in eine oder mehrere weitere.

Tabelle 3-8: Spezifische Parameter der Energieumwandlung für *Tessif* (*Transformer*) und *FINE* (*Conversion*). *als *ConversionDynamic* Komponente angelegt für *FINE*.

<i>Tessif</i>	<i>FINE</i>	Beschreibung
<i>name</i>	<i>name</i>	Benennung
<i>carrier</i>	<i>commodity</i>	Energieträger
<i>timeseries</i>	<i>operationRateMax</i>	Erzeugungsprofil (Daten)
<i>flow_rates</i>	<i>capacityMin</i>	Minimale Leistung
<i>expansion_limit</i>	<i>capacityMax</i>	Maximal Leistung
<i>flow_cost</i>	<i>opexPerOperation</i>	Betriebskosten (Opex)
<i>expansion_cost</i>	<i>investPerCapacity</i>	Investitionskosten (Capex)
<i>conversion</i>	<i>commodityConversionFactors</i>	Umwandlungsfaktoren
<i>flow_emissions</i>		
<i>flow_gradients</i>	<i>rampUpMax*</i> <i>rampDownMax*</i>	Änderungsrate der Leistung

Technisch relevant ist der Wirkungsgrad, um verlustbehaftete Umwandlungen zu definieren. Installierte Leistungen, Emissionen und Kosten dieser Komponente sind in *FINE* auf den Aufwand (*Inflow*) bezogen. Deshalb wird mittels des Wirkungsgrades

$$\text{Wirkungsgrad} = \frac{\text{Nutzen}}{\text{Aufwand}} = \frac{\text{Outflow}}{\text{Inflow}} \quad (3.1)$$

eine Umrechnung integriert. Die Leistung muss dementsprechend durch den Wirkungsgrad geteilt werden, wohingegen Kosten und Emissionen mit diesem multipliziert werden. Entstehen mehrere Energieformen durch die Umwandlung, müssen außerdem die jeweiligen Anteile an den Kosten und Emissionen auf den *Inflow* bezogen und aufsummiert werden. Hierdurch ist die eingangs beschriebene lokale Erweiterung des *FINE* ES notwendig, um später die spezifischen Werte rekonstruieren zu können. Die in *Tessif* separaten spezifischen Emissionen (*flow_emissions*) liegen in *FINE* in einem Parameter mit

dem Wirkungsgrad (*commodityConversionFactors*) bezeichnet als Umwandlungsfaktoren. Zusätzlich kann mit Hilfe der dynamischen Energieumwandlung in *FINE* der *Tessif* Parameter bezüglich der Laständerungsrate (*flow_gradients*) zur Begrenzung übersetzt werden.

Die Speicherkomponente (*Storage*) ist erneut in *Tessif* und *FINE* gleich benannt und die zu übersetzenden Parameter können der folgenden Tabelle 3-9 entnommen werden.

Tabelle 3-9: Spezifische Parameter des Speichers für *Tessif* (*Storage*) und *FINE* (*Storage*).

<i>Tessif</i>	<i>FINE</i>	Beschreibung
<i>name</i>	<i>name</i>	Benennung
<i>carrier</i>	<i>commodity</i>	Energieträger
<i>flow_efficiencies</i>	<i>chargeEfficiency</i> <i>dischargeEfficiency</i>	Lade-/ Entladewirkungsgrade
<i>capacity</i>	<i>capacityMin</i>	Vorhandene Kapazität
<i>idle_changes</i>	<i>selfDischarge</i>	Selbstentladung
<i>flow_rates</i>	<i>chargeRate</i>	Lade-/ Entladeleistung
<i>flow_gradients</i>	<i>dischargeRate</i>	
<i>flow_rates</i> <i>expansion_limit</i>	<i>capacityMax</i>	Maximale Kapazität
<i>flow_cost</i>	<i>opexPerDischargeOperation</i>	Betriebskosten (Opex)
<i>expansion_cost</i>	<i>investPerCapacity</i>	Investitionskosten (Capex)

Zunächst wurde die vorhandene Kapazität (*capacity*) zu der minimalen Kapazität (*capacityMin*) in *FINE* übersetzt. Weitere technische Parameter beschreiben das Lade- und Entladeverhalten des Speichers. Hierzu dient der Wirkungsgrad beim Laden (*chargeEfficiency*) und Entladen (*dischargeEfficiency*) sowie die Selbstentladung (*selfDischarge*) des Speichers pro Zeitschritt. Begrenzt werden kann die maximale Ein- und Ausspeiseleistung, analog zu den Lastwechselraten der Energieumwandlungen oder mit Hilfe der maximalen Energieflussrate (*flow_rates*).

Als letzte Komponente müssen abschließend Grenzwerte nach Tabelle 3-10 in *FINE* über eine Senke definiert werden, welche die Restriktion bezüglich der jeweiligen Ressource oder Emission als jährliches Limit (*yearlyLimit*) festlegt.

Tabelle 3-10: Spezifische Parameter der Grenzwerte in *Tessif* (*global_constraints*) und *FINE* (*Sink*).

<i>Tessif</i>	<i>FINE</i>	Beschreibung
<i>name</i>	<i>name</i>	Benennung
<i>carrier</i>	<i>commodity</i>	Emission/Ressource
/	<i>yearlyLimit</i>	Grenzwert
/	<i>commodityLimitID</i>	Identifikation des Grenzwertes

Übernommen und benannt werden können diese durch die in *Tessif* vorhandenen Begrenzungen (*global_constraints*). Das jährliche Limit wird in *FINE* immer auf die Zeitspanne von einem Jahr bezogen, weshalb eine Flexibilisierung dieser in der *es2es*-Funktion möglich ist. Die Identifikation des Grenzwertes (*commodityLimitID*) muss angelegt werden, sobald ein jährliches Limit erstellt wird, damit der Import bzw. Export von mehreren Komponenten in Summe begrenzt werden kann. Sollte die Verwendung eines Energieträgers begrenzt sein (*accumulated_amounts*), werden die Parameter der jeweiligen Komponente aus den vorherigen Tabellen mit der individuellen Identifikation erweitert. Zusätzlich werden sie mit dem erlaubten jährlichen Limit auf die Länge der Zeitspanne skaliert.

Mit der Definition von Restriktionen ist die Integration der Basiskomponenten die erforderlich sind, um für *FINE* ein Energiesystem zu erstellen abgeschlossen. *Tessif* hingegen verfügt über weitere Komponenten, die nicht direkt durch *FINE* vorgegeben sind. Die Integration dieser, mittels Kombinationen aus Basiskomponenten, wird in dem folgenden Kapitel näher erläutert.

3.2.1.2 Integration der indirekten Komponenten

Wie zu Beginn angeführt enthält *Tessif* zusätzliche Komponenten und Eigenschaften dieser, die nicht unmittelbar in *FINE* vorhanden sind. Um einen erweiterten Nutzen durch *Tessif* zu erzielen, wird im Folgenden die spezielle Entwicklung dieser Komponenten erläutert.

Ein Konnektor dient dazu, Potentialebenen miteinander zu verbinden und ermöglicht so Energieträger direkt zwischen den transportierenden Komponenten zu übermitteln. Er kann, wie in der folgenden Abbildung 3-3 dargestellt, formuliert werden.

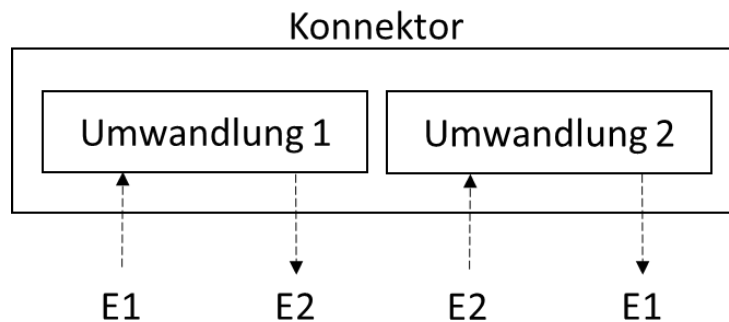


Abbildung 3-3: Schematische Darstellung des Konnektors in *Tessif* entwickelt für *FINE* durch Kombination von Energieumwandlungen.

Die Kombination von zwei Umwandlungen aus *FINE* zu einer indirekten Komponente ermöglicht es die Energieträger (E1, E2) in jeweils eine Richtung ineinander zu wandeln. Dadurch ist es beispielsweise möglich, verschiedene Spannungsebenen miteinander zu verbinden. Alle erforderlichen Parameter sind in Tabelle 3-11 aufgeführt und werden durch die Transformation doppelt erzeugt und direkt einander zugeordnet.

Tabelle 3-11: Spezifische Parameter des Konnektors in *Tessif* entwickelt für *FINE* durch Kombination von Energieumwandlungen.

<i>Tessif</i>	<i>FINE</i>	Beschreibung
<i>name</i>	<i>name</i>	Benennung
<i>carrier</i>	<i>commodity</i>	Energieträger
<i>conversion</i>	<i>commodityConversionFactors</i>	Umwandlungsfaktoren

Eine weitere indirekt zu formulierende Komponente ist eine emissionsbehaftete Quelle. In *FINE* können Emissionen dieser nicht direkt beigemessen werden, weshalb wieder eine Kombination entwickelt wird. Soll Energie durch eine emittierende Quelle erzeugt werden, so wird gleichzeitig eine fiktive und unbegrenzte Quelle erstellt, die nicht Teil des ES ist. Diese versorgt eine Energieumwandlung, die als eigentliche Quelle angesehen werden kann. Wie in Abbildung 3-4 zu erkennen, entstehen die spezifischen Emissionen bei der verlustfreien Umwandlung der fiktiven in die systeminterne Energie.

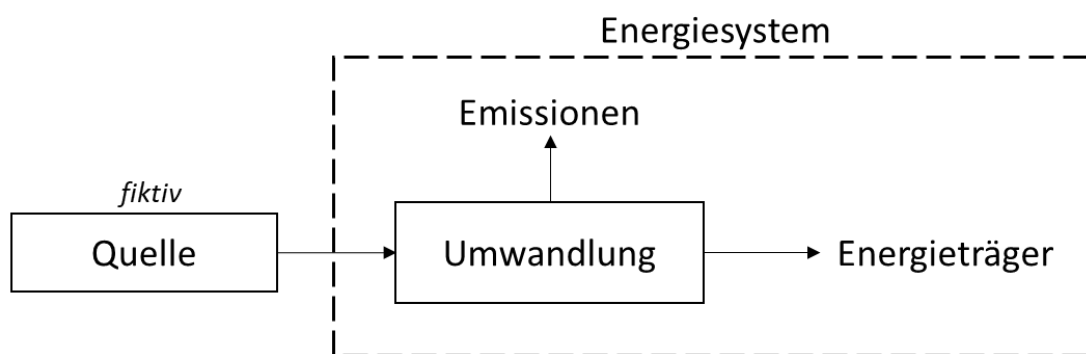


Abbildung 3-4: Schematische Darstellung der emittierenden Quelle entwickelt für *FINE* durch Kombination einer fiktiven Quelle und einer Energieumwandlung.

Für die Integration ist zu beachten, dass der Energieträger der fiktiven Quelle von allen weiteren im System enthaltenen Energieträgern eindeutig abgegrenzt werden muss. So wird sichergestellt, dass diese unbegrenzt zur Verfügung stehende Ressource ausschließlich durch die zugehörige Umwandlung in Anspruch genommen werden kann. Alle flussspezifischen Parameter der Umwandlung werden analog zu denen einer Quelle (Tabelle 3-7) ermittelt und zusätzlich mit dem verlustfreien Wirkungsgrad und den spezifischen Emissionen in den Umwandlungsfaktoren (*commodityConversionFactors*) erweitert. Die zugehörigen zu übersetzenden Parameter können Tabelle 3-12 entnommen werden.

Tabelle 3-12: Spezifische Parameter der emittierenden Quelle entwickelt für *FINE* durch Kombination einer fiktiven Quelle und einer Energieumwandlung.

<i>Tessif</i>	<i>FINE</i>	Beschreibung
<i>name</i>	<i>name</i>	Benennung (Quelle)
<i>carrier</i>	<i>commodity</i>	Energieträger (Quelle)
<i>name</i>	<i>name</i>	Benennung (Umwandlung)
<i>carrier</i>	<i>commodity</i>	Energieträger (Umwandlung)
<i>timeseries</i>	<i>operationRateMax</i>	Erzeugungsprofil (Daten)
<i>conversion</i>	<i>commodityConversionFactors</i>	Umwandlungsfaktoren
<i>flow_rates.min</i> <i>expansion_limit.min</i>	<i>capacityMin</i>	Minimale Leistung
<i>flow_rates.max</i> <i>expansion_limit.max</i>	<i>capacityMax</i>	Maximale Leistung
<i>flow_cost</i>	<i>opexPerOperation</i>	Betriebskosten (Opex)
<i>expansion_cost</i>	<i>InvestPerCapacity</i>	Investitionskosten (Capex)

Die dritte und letzte indirekt zu integrierende Komponente sind emissionsbehaftete Speicher. Die Kombination von Komponenten erfolgt hier wieder durch die Kombination mit verlustfreien Energieumwandlungen. Nach Abbildung 3-5 wird der zu speichernde Energiefluss umgewandelt in eine allein dem Speicher zuzuordnende Energieform.

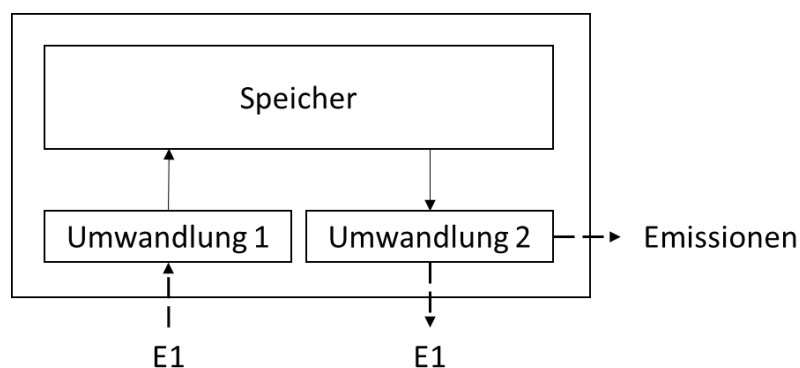


Abbildung 3-5: Schematische Darstellung eines emittierenden Speichers als indirekte Komponente entwickelt für *FINE* durch Kombination mit zwei Energieumwandlungen.

Die spezifischen Emissionen entstehen während des Entladevorgangs. Sobald die Energie dem System wieder zugeführt werden soll, wird diese durch die zweite Umwandlung ver-

lustfrei in die im System vorhandene Energieform überführt. Mit den Parametern aus Tabelle 3-13 ist somit die letzte indirekte Komponente als emittierender Speicher aus *Tessif* für *FINE* verfügbar.

Tabelle 3-13: Spezifische Parameter eines emittierenden Speichers als indirekte Komponente entwickelt für *FINE* durch Kombination mit zwei Energieumwandlungen. *eindeutig zuzuordnen.

<i>Tessif</i>	<i>FINE</i>	Beschreibung
<i>name</i>	<i>name</i>	Benennung (Umwandlung)*
<i>carrier</i>	<i>commodity</i>	Energieträger*
<i>conversion</i> <i>flow_emissions</i>	<i>commodityConversionFactors</i>	Umwandlungsfaktoren
<i>name</i>	<i>name</i>	Benennung (Speicher)
<i>carrier</i>	<i>commodity</i>	Energieträger
<i>flow_efficiencies</i>	<i>chargeEfficiency</i> <i>dischargeEfficiency</i>	Lade-/ Entladewirkungsgrade
<i>capacity</i>	<i>capacityMin</i>	Vorhandene Kapazität
<i>idle_changes</i>	<i>selfDischarge</i>	Selbstentladung
<i>flow_gradients</i>	<i>chargeRate</i> <i>dischargeRate</i>	Lade-/ Entladeleistung
<i>flow_rates</i> <i>expansion_limit</i>	<i>capacityMax</i>	Maximale Kapazität
<i>flow_cost</i>	<i>opexPerDischargeOperation</i>	Betriebskosten (Opex)
<i>expansion_cost</i>	<i>investPerCapacity</i>	Investitionskosten (Capex)

Alle durch die *es2es*-Funktion übersetzten Komponenten müssen nun im Folgenden geprüft werden, sodass eine korrekte Integration für eine anschließende detaillierte Szenarioanalyse bestätigt werden kann.

3.2.1.3 Prüfen der integrierten Komponenten

In *Tessif* wurden bereits vereinfachte Beispiele gemäß Tabelle 3-14 zur Modellierung und Optimierung von kleinen ES entwickelt, um die Funktionsweise einzelner Komponenten zu erläutern und zu untersuchen. Anhand dieser wurden die jeweils transformierten Parameter mittels Simulation und Optimierung kontinuierlich geprüft. Jedes Beispiel untersucht eine kurze Zeitspanne, innerhalb derer ein definierter Bedarf gedeckt werden soll. Grundsätzlich galt es, die sechs Komponenten aus *Tessif* zu integrieren, wobei Quellen und Senken Teil jedes ES sind. Zusätzlich müssen Emissionsrestriktionen geprüft und ein unbegrenzter Ausbau untersucht werden. Anhand der Beispiele ist es somit möglich das Verhalten der einzelnen Komponenten zu untersuchen und gleichzeitig zu erforschen, wie sich verschiedene Parameter auswirken.

Tabelle 3-14: Beispiel Energiesysteme für die Basiskomponenten in *Tessif* und *FINE*.

Energiesystem	Beschreibung
<i>Minimum Working Example</i>	Verbindung von Komponenten
<i>Emission Objective</i>	Begrenzung der Emissionen
<i>CHP Example</i>	Einsatz einer Energieumwandlung
<i>Storage Example</i>	Einsatz eines Speichers
<i>Connected Example</i>	Einsatz eines Konnektors
<i>Expansion Example</i>	Ausbau von Komponenten

Die Bilanzierung von weniger komplexen Energiesystemen sollte ein eindeutiges Ergebnis erzielen. Da für jedes übersetzte Beispiel *Oemof*, *PyPSA* und *FINE* die gleichen Ergebnisse erzielen, kann vorläufig von der korrekten Integration *FINE's* ausgegangen werden. Eine detaillierte Überprüfung der jeweilig erzielten Optimierungsergebnisse soll im Verlauf dieser Arbeit anhand eines komplexen Energiesystems durchgeführt werden.

3.2.2 Energy-System to Mapping (*es2mapping*)

Jedes Softwaremodell zeichnet sich durch seine eigene Logik bei der Datenspeicherung aus. Daher können die Ergebnisse des optimierten ES nicht direkt für den Vergleich mit den Resultaten anderer Softwaremodelle herangezogen werden. Zu diesem Zweck ist eine standardisierte Darstellung erforderlich, die in *Tessif* für *Oemof* und *PyPSA* bereits existiert und für *FINE* im Verlauf dieser Arbeit entwickelt werden soll.

Um die standardisierte Darstellung zu erreichen, wird zunächst die Struktur des optimierten *FINE* ES analysiert. Dabei wird identifiziert, welche Ergebnisse relevant für die spätere Datenauswertung sind und ob diese gegebenenfalls vorher umgerechnet werden müssen. Die *es2mapping*-Funktion wird in vier Kategorien unterteilt, die jeweils spezifische Informationen zusammenstellen. Die ersten drei Kategorien sind technisch und betrachten Erzeugungs- und Lastprofile (*Loads*), installierte Leistungen (*Capacities*) sowie flussbezogene Größen zu Kosten und Emissionen (*Flows*). Die vierte Kategorie bezieht sich dann auf die ökonomischen und ökologischen Gesamtergebnisse (*Global Results*). Diese setzen sich aus den Gesamtkosten und den insgesamt entstehenden Emissionen zusammen. Die Gesamtkosten werden zudem unterteilt in Betriebs- und Investitionskosten. Weiterhin wird die Funktion so entwickelt, dass mit Hilfe der *UID*'s, eine effektive Formatierung der Ergebnisse möglich ist. Mit den beinhalteten Informationen gemäß Tabelle 3-2, zu beispielsweise Energieträgern oder zur sektoralen Zugehörigkeit, soll ein schneller Zugang zu relevanten Daten ermöglicht werden.

Im Folgenden wird das Vorgehen während der Entwicklung der vier Kategorien für die standardisierte Ergebnisdarstellung erläutert.

3.2.2.1 Erzeugungs- und Lastprofile (*Loads2mapping*)

Durch die Optimierung der Komponenten wird der zeitliche Verlauf für erzeugte Energie festgelegt. Bei den Verbrauchern ist grundsätzlich ein festes Profil vorgegeben, das durch die erzeugenden Komponenten möglichst effektiv gedeckt werden soll. Um die ermittelten Lastprofile von Softwaremodellen untereinander vergleichbar zu machen, werden sie von *Tessif* vereinheitlicht. Auf dieser Basis können anschließend alle ein- und austreten-

den Energieflüssen einer Komponente zusammengestellt werden. Dafür wird grundsätzlich festgelegt, dass eintretende Energieflüsse ein negatives und austretende ein positives Vorzeichen erhalten. Quellen und Senken weisen lediglich einen Energiefluss auf, wohingegen Umwandlungen einen Energieträger aufnehmen und diesen in einen oder mehrere weitere umwandeln. Mit Hilfe des Wirkungsgrads nach Gleichung (3.1) ist es möglich alle relevanten Daten zu rekonstruieren, da die durch *FINE* ermittelten Ergebnisse auf den *Inflow* bezogen sind. Der durch *FINE* generierte *Inflow* wird daher multipliziert mit dem Wirkungsgrad, um den *Outflow* zu berechnen.

Es entsteht für jede Komponente eine Tabelle, welche alle spezifischen Energieflüsse enthält. Die Beschriftung der Spalten umfasst zusätzlich die Information, von welcher oder zu welcher anderen Komponente der jeweilige Energiefluss transportiert wird.

Tabelle 3-15: Vorzeichendefinition eintretende Ströme negativ, austretende positiv einer Energieumwandlung über drei Zeitschritte.

Zeitschritt	Quelle	Ziel 1	Ziel 2
1	-	+	+
2	-	+	+
3	-	+	+

Um diese Form generieren zu können, bedarf es der Information, welche Komponente untersucht wird und mit welchen Weiteren sie kommuniziert. *FINE* verfügt hierfür über spezifische Parameter, die genutzt werden, um die *es2mapping*-Funktion zu entwickeln. Angefangen wird mit der Dimension der jeweiligen Komponente die beschreibt, ob Energieflüsse uni- oder bidirektional sind. Energietransporte werden zweidimensional (*2dim*) und alle weiteren Komponenten eindimensional (*1dim*) beschrieben. Erzeugende eindimensionale Komponenten können ihren Energieträger ausschließlich in zweidimensionale Energietransporte mit dem gleichen Energieträger einspeisen. Umgekehrt führen die Ergebnistabellen zu den Energietransporten demnach alle Erzeuger mit gleichem Energieträger auf.

Quellen und Senken sind zudem mit einem Vorzeichenparameter (*sign*) versehen. Dieser beschreibt, ob ein Energieträger dem ES zugeführt (+1) oder entnommen (-1) wird. Diese Information wird genutzt, um Komponenten, die den gleichen Energieträger (*com*) besitzen, dem jeweiligen Energietransport mit dem korrekten Vorzeichen zuzuordnen. Durch

FINE wird der Energieumwandlungen kein spezieller Energieträger zugeordnet, da mindestens zwei verwendet werden. Energieumwandlungen und Speicher weisen keinen Vorzeichenparameter auf, weshalb diese gesondert betrachtet werden müssen. Unterschieden werden kann durch den Energieträger, der bei Speichern vorhanden ist, bei Energieumwandlungen hingegen nicht.

Zur Erläuterung werden in der folgenden Tabelle beispielhaft die drei *FINE*-spezifischen Parameter für das in Kapitel 2.5.2 Abbildung 2-9 untersuchte ES aufgeführt.

Tabelle 3-16: *FINE* spezifische Parameter Dimension, Vorzeichen und Energieträger für die eindeutige Beschreibung der Komponenten des in Abbildung 2-3 und Abbildung 2-9 dargestellten Energiesystems.

Komponente	<i>dim</i>	<i>sign</i>	<i>com</i>
Stromnetz	<i>2dim</i>	-	<i>Strom</i>
Wärmenetz	<i>2dim</i>	-	<i>Wärme</i>
Strombedarf	<i>1dim</i>	-1	<i>Strom</i>
Wärmebedarf	<i>1dim</i>	-1	<i>Wärme</i>
Gasimport	<i>1dim</i>	1	<i>Gas</i>
Ersatzstrom	<i>1dim</i>	1	<i>Strom</i>
Ersatzwärme	<i>1dim</i>	1	<i>Wärme</i>
BHKW	<i>1dim</i>	-	-
Emissionslimit	<i>1dim</i>	-1	<i>CO₂</i>
Batterien	<i>1dim</i>	-	<i>Strom</i>

Das untersuchte ES enthält weiterhin eine Restriktion zu Emissionen, die als Senke aufgeführt wird. Diese kann wieder anhand des Energieträgers identifiziert werden, sodass keine Komplikationen während des Zusammenstellens der Lasten auftreten. Mit Hilfe dieser drei Parameter ist es möglich, jede Komponente eindeutig dem entsprechenden Energietransport zuzuordnen und ihre Abhängigkeiten zu definieren. So können die Erzeugungs- und Lastprofile für jeden spezifischen Energiefluss einer Komponente in der entsprechenden Darstellung kombiniert werden.

3.2.2.2 Installierte Leistungen und Kapazitäten (*Capacities2mapping*)

Installierte Leistungen für Erzeuger und Verbraucher bzw. Kapazitäten der Speicher können einen schnellen Überblick über die Ergebnisse einer Optimierung ermöglichen. Sie erlauben es darüber hinaus, die berechneten Investitionskosten zu überprüfen. In *FINE* werden diese Werte nach Kapitel 3.2.1 als minimale und maximale Leistung bzw. Kapazität für jede Komponente angelegt, weshalb diese durch die *es2mapping*-Funktion aus dem optimierten ES ausgelesen und einheitlich zusammengestellt werden.

Bei Energieumwandlungen muss erneut berücksichtigt werden, dass die durch *FINE* ermittelten Werte auf den Zufluss bezogen sind und die standardisierte Form von *Tessif* vorsieht, dass alle Werte den entstehenden Energiefluss repräsentieren. Mit Hilfe von Gleichung (3.1) für den Wirkungsgrad wird deshalb eine automatisierte Umrechnung integriert, um vergleichbare Ergebnisse zu schaffen.

Der minimal installierte Leistungsparameter (*capacityMin*) wird durch die Integration als die bereits vorhandene Leistung interpretiert. Dies ist von Bedeutung, wenn das ES nicht auf dem Ansatz der „Grünen Wiese“ basiert, sondern bereits vorhandene Leistungen bzw. Kapazitäten existieren. Bei der Berechnung der Gesamtkosten in Kapitel 3.2.2.4 muss dies berücksichtigt werden, da sonst höhere Investitionskosten durch *FINE* entstehen.

3.2.2.3 Flussbezogene Ergebnisse (*Flows2mapping*)

Die flussbezogenen Ergebnisse beinhalten für jede Komponente spezifische Emissionen, Kosten sowie insgesamt umgesetzte Energiemengen. Zugeordnet werden die Ergebnisse sogenannten Verbindungen bzw. Kanten (*Edges*), die exakt definieren, auf welchen Ursprungs- und Zielknoten sich die jeweiligen Werte beziehen. Die Ausgangsbasis bilden erneut die drei durch *FINE* definierten Parameter zur Dimension, dem Vorzeichen und der Energieform. Es wird davon ausgegangen, dass jede eindimensionale Komponente auf einen zweidimensionalen Energietransport mit dem gleichen Energieträger zeigt. Dadurch entsteht die Kante

$$Edge (Ursprung, Energietransport) = flussbez. Größe \quad (3.2)$$

für jede erzeugende Komponente mit der zugehörigen spezifischen Größe. Entstehen durch eine Energieumwandlung zwei oder mehr Energieflüsse, so kann pro Fluss jeweils eine Kante ermittelt werden. Ursprünge können demnach Quellen, Energieumwandlungen sowie Speicherentladungen sein. Umgekehrt stellt auch jeder Energietransport eine zugehörige Kante

$$\textit{Edge} (\textit{Energietransport}, \textit{Ziel}) = \textit{flussbez. Größe} \quad (3.3)$$

dar, der jedoch auf die verbrauchenden Komponenten zeigt. Einem Energietransport wird durch *Tessif* kein spezifischer Wert zugeordnet, jedoch werden die Kanten für die spätere Visualisierung benötigt. Mögliche Zielknoten sind Senken, die Zuflüsse von Umwandlungen und Speicherladungen mit jeweils gleichen Energieträgern wie der Energietransport. Die jeweiligen Emissionen und Kosten können den Komponenten des ES direkt entnommen werden. Diese stellen dann den Ursprungsknoten für die Kante dar und über den Energieträger kann der zugehörige zweidimensionale Energietransport ermittelt werden. Die insgesamt umgesetzten Energieflüsse können unter Hinzunahme der bereits erstellen Erzeugungs- und Lastprofile (Kapitel 3.2.2.1) ermittelt werden. Dafür wird das jeweilige Profil eines Energieflusses aufsummiert und der passenden Kante zugeordnet.

Mit den Verbindungen bzw. Kanten ist es möglich, jeden Energiefluss eindeutig zu beschreiben und die flussbezogenen Ergebnisse darzustellen. Somit ist eine wichtige Grundlage für einen detaillierten komponentenbasierten Softwarevergleich geschaffen.

3.2.2.4 Gesamtergebnisse (*Global Results*)

Die durch *Tessif* standardisiert dargestellten Gesamtergebnisse beziehen sich sowohl auf die Gesamtkosten als auch auf die entstandenen Emissionen. Grundsätzlich soll durch eine Optimierung prognostiziert werden, wie viele Emissionen für ein bestimmtes ES entstehen und welche Kosten für Veränderungen notwendig sind. Aufgrund dessen sind die Emissionen ein essenzieller Teil der Gesamtergebnisse. Wie in Kapitel 3.2.1.2 erläutert, werden diese abstrakt als Senke beschrieben, weshalb in *FINE* ein Erzeugungsprofil entsteht. Die Summe über alle Zeitschritte entspricht schließlich den insgesamt entstandenen Emissionen für das untersuchte ES. Auf Plausibilität geprüft werden können diese mit Hilfe der nach Kapitel 3.2.2.3 ermittelten flussspezifischen Ergebnisse.

$$Emissionen_{ges}^{FINE} \stackrel{!}{=} \sum_{K=1}^n (P_{K,ges} \cdot spez. Emissionen_K) \quad (3.4)$$

Dafür wird die gesamt umgesetzte Energiemenge $P_{K,ges}$ einer Komponente K multipliziert mit den spezifischen Emissionen und für alle n Komponenten aufsummiert. Entnommen werden die Emissionen den Umwandlungsfaktoren der jeweiligen Energieumwandlung aus Tabelle 3-8.

Aufgrund des übergeordneten Ziels der Kostenminimierung, werden während jeder Optimierung durch ein Softwaremodell die Gesamtkosten (*objectiveValue*) berechnet. Für *FINE* beinhalten die Investitionskosten die bereits vorhandenen Anteile der Komponenten. In Kapitel 3.2.1 wurde dargelegt, dass die vorhandene Leistung (*capacityMin*) einer Komponente, den durch *Tessif* definierten bereits installierten Leistungen entspricht. *FINE* interpretiert diesen Parameter als mindestens zu installierenden Anteil einer Komponente und berücksichtigt die Investitionen

$$CAPEX_{inst.} = \sum_{K=1}^n (capacityMin \cdot investPerCapacity)_K \quad (3.5)$$

in den Gesamtkosten. Die Berechnung erfolgt mittels der Summe aller bereits installierten Anteile (*capacityMin*) multipliziert mit den spezifischen Investitionskosten (*investPerCapacity*) für alle Komponenten n . Um die in *Tessif* vergleichbaren Kosten

$$Kosten_{ges}^{FINE} = objectiveValue - CAPEX_{inst.} \quad (3.6)$$

zu berechnen, müssen die Investitionen der bereits vorhandenen Leistungen und Kapazitäten ($CAPEX_{inst.}$) von den durch *FINE* ermittelten Gesamtkosten (*objectiveValue*) subtrahiert werden. Anhand dessen entsteht die direkte Bedingung

$$Kosten_{ges}^{FINE} \stackrel{!}{=} OPEX_{ges}^{Tessif} + CAPEX_{ges}^{Tessif} \quad (3.7)$$

für die *Tessif* Ergebnisse. Die errechneten Betriebs- (*OPEX*) und Investitionskosten (*CAPEX*) sollten dabei den ermittelten Gesamtkosten durch das Softwaremodell entsprechen. Die können einzeln anhand der zusammengestellten Ergebnisse bestimmt werden, um eine detailliertere Untersuchung und Plausibilitätsprüfung zu ermöglichen.

Für die operationellen Kosten

$$OPEX_{ges}^{Tessif} = \sum_{K=1}^n (P_{K,ges} \cdot opexPerOperation_K) \quad (3.8)$$

wird erneut die pro Zeitschritt erzeugte Leistung aus dem Erzeugungsprofil der Komponente K aufsummiert zu der insgesamt umgesetzten Leistung $P_{K,ges}$ aller n Komponenten. Der jeweilige Wert wird dann multipliziert mit den spezifischen Kosten pro Energiefluss. Die Summe aus allen Komponenten ergibt dann die gesamten für das ES und den Zeitraum erforderlichen operationellen Kosten.

Die Investitionskosten

$$CAPEX_{ges}^{Tessif} = \sum_{K=1}^n (capacityMax \cdot investPerCapacity)_K \quad (3.9)$$

werden berechnet, indem die installierte Leistung bzw. Kapazität der Komponente K mit den spezifischen Investitionskosten multipliziert und für alle n Komponenten aufsummiert wird.

Die standardisierte Darstellung der Gesamtergebnisse bezüglich Emissionen und Kosten des untersuchten ES finalisiert die *es2mapping*-Funktion. Dadurch ist die Basis geschaffen, auf der ein analytischer Vergleich der Ergebnisse von Software zur Modellierung und Optimierung von ES folgen kann.

3.3 Auswahl und Beschreibung des Beispielenergiesystems

Um die Integration von *FINE* zu prüfen und erste Ergebnisse vergleichen zu können, muss ein geeignetes Energiesystem ausgewählt werden. *Tessif* stellt eine Bibliothek an Systemen, die speziell für einen Vergleich von Software entwickelt wurden.

Das von Reimer [41] entwickelte komponentenbasierte Energiesystem (*component_es*) ermöglicht eine Differenzierung in Referenz- und Basisszenario für die Untersuchung. Weiterhin zeichnet es sich durch eine übersichtliche Architektur des Systems, bei gleichzeitiger detaillierter und realistischer Beschreibung relevanter Parameter aus. Die Komponenten sind übergeordnet gruppiert und ermöglichen so eine unkomplizierte Analyse der Sektoren Strom und Wärme sowie der Vorgehensweise, wie das jeweilige Softwaremodell die Energieflüsse durch Komponenten erzeugt. Aus diesen Gründen wird das komponentenbasierte Energiesystem ausgewählt, um eine erste vergleichende Analyse der neu integrierten Software *FINE* mit den bereits vorhandenen *Oemof* und *PyPSA* durchzuführen. Im Folgenden wird auf die Ziele der beiden Szenarien sowie auf die Zusammenstellung der ausgewählten Komponenten eingegangen.

Alle grundlegenden Spezifikationen der beiden Szenarien Referenz und Basis werden in Tabelle 3-17 aufgeführt.

Tabelle 3-17: Spezifikationen der Szenarien des *component_es* der *Tessif* Bibliothek.

Szenario	Ausbau	Restriktion
Referenz	nicht möglich	keine
Basis	möglich	250.000 $\frac{\text{t CO}_2}{\text{Jahr}}$

Durch das Referenzszenario (*Commitment*) werden für eine Einsatzplanoptimierung, wie in Kapitel 2.2 beschrieben, entstehende Kosten und Emissionen ermittelt ohne, dass eine Veränderung am Energiesystem selber vorgenommen werden darf. Daher ist fest vorgegeben, wie viel installierte Leistung bzw. Kapazität im System vorhanden ist. Da keine Restriktionen bezüglich Emissionen oder Ressourcen vorgegeben sind, werden Entscheidungen bezüglich der Erzeugung rein ökonomisch getroffen. In dem Basisszenario (*Ex-*

pansion) hingegen ist ein Ausbau aller im System befindlichen, erzeugenden Komponenten möglich, um ein vorgegebenes Emissionslimit einzuhalten. Ressourcen sind dagegen keiner Restriktion unterworfen und können, wie im Referenzszenario, unbegrenzt verwendet werden. Durch die Optimierung soll dann ermittelt werden, wie das ES angepasst werden muss, um das Emissionsziel einzuhalten.

Um die beiden Szenarien mittels unterschiedlicher Software zu untersuchen, wurde ein abstraktes, aber realitätsnahes ES entwickelt, dessen Komponenten im Folgenden betrachtet werden sollen. Allgemein werden die gekoppelten Sektoren Strom und Wärme dargestellt und es kann geprüft werden, wie die jeweilige Software mit den Energieformen und Nachfragen umgeht. Die in Kapitel 2.6.1 eingeführten Komponenten werden, bis auf den Konnektor, alle in diesem Beispiel berücksichtigt. Der Ausschluss von Konnektoren ist darauf zurückzuführen, dass die Energietransporte idealisiert betrachtet werden, weshalb keine Differenzierung in Potentialebenen (z.B. Spannungen, Drücke) erforderlich ist.

Die verbrauchenden Komponenten sind übergeordnet zusammengefasst in jeweils eine Senke pro Sektor. Sie repräsentieren den deutschen Bedarf von Strom für 2019 und Wärme für 2013, jeweils skaliert auf eine Stadt der Größe Hamburgs. Die zugehörigen Werte können in Tabelle 3-18 gefunden werden.

Tabelle 3-18: Strom- und Wärmebedarf für das *component_es* der *Tessif*-Bibliothek.

Sektor	Bedarf / TWh
Strom	9,81
Wärme	1,12

Für beide ist ein festes Lastprofil vorgeschrieben, welches durch die zur Verfügung stehenden Technologien gedeckt werden muss. Die Quellen erneuerbarer Energien (RES) beinhalten Photovoltaik (PV), Windkraftanlagen (WKA) On- und Offshore sowie ein Biogas Blockheizkraftwerk (BHKW). Während durch PV und WKA ausschließlich Strom erzeugt wird, können durch das Biogas-BHKW gleichzeitig Strom und Wärme bereitgestellt werden. Vorgegeben sind für die fluktuierenden RES realistische Erzeugungsprofile anhand von Wetterdaten, die auf den jeweilig installierten Anteil normiert sind, um die Erzeugung

während der Optimierung zu begrenzen. Die bereits installierten Leistungen des ES wurden weiterhin von den Anteilen der installierten Nettoleistung Deutschlands abgeleitet und auf die Größe des ES skaliert.

Darüber hinaus emittieren die RES weniger Emissionen als fossile Erzeuger. Während die Emissionen für das Biogas-BHKW bei der Energieumwandlung entstehen, sind für PV und Windkraft, jene die während des Lebenszyklus entstehenden. Die Emissionen der fossilen Erzeuger entstehen vorwiegend durch die Verbrennung des Brennstoffes. Der Mix aus fossilen Energien besteht aus Braun (BK)- und Steinkohle (SK) Kraftwerken sowie einem Gas- und Dampfkraftwerk (GuD-Kraftwerk). Während die Kraftwerke ausschließlich zur Strombereitstellung dienen, kann ein zusätzliches SK-Kraftwerk auch gekoppelt Strom- und Wärme (KWK) erzeugen. Dimensioniert sind die fossilen Erzeuger so, dass die notwendige Spitzenlast des Stromsektors allein durch sie gedeckt werden kann.

Wird der Wärmesektor genauer betrachtet, ist zu erkennen, dass dieser eine geringere Anzahl an Komponenten umfasst. Neben der bereits erwähnten thermischen Erzeugung durch Steinkohle-KWK und das Biogas-BHKW, kann der Wärmebedarf zusätzlich durch ein Heizwerk sowie durch Power-to-Heat (PtH) gedeckt werden, wobei die Spitzenlast wieder allein durch das Heizwerk abgedeckt werden kann. Die PtH-Anlage ermöglicht die anteilige Elektrifizierung des Wärmesektors und dient als eine Verbindung für die Koppelung der beiden Sektoren.

Um den Einfluss und die Handhabung saisonaler Speicher durch die Software zu testen, enthält das ES weiterhin für beide Energieträger jeweils einen Energiespeicher. Diese sollen Energie umwandeln und je nach Überschuss speichern, um eine intelligente Nutzung zu ermöglichen und Fluktuationen der RES auszugleichen. Damit eine realistische Nutzung der Speicher durch die Software gewährleistet ist, sind Batterie und Wärmespeicher jeweils mit einer Selbstentladerate von 0,5 % versehen.

Die Modellarchitektur des ES wird in Abbildung 3-6 mittels *Tessif* visualisiert dargestellt.

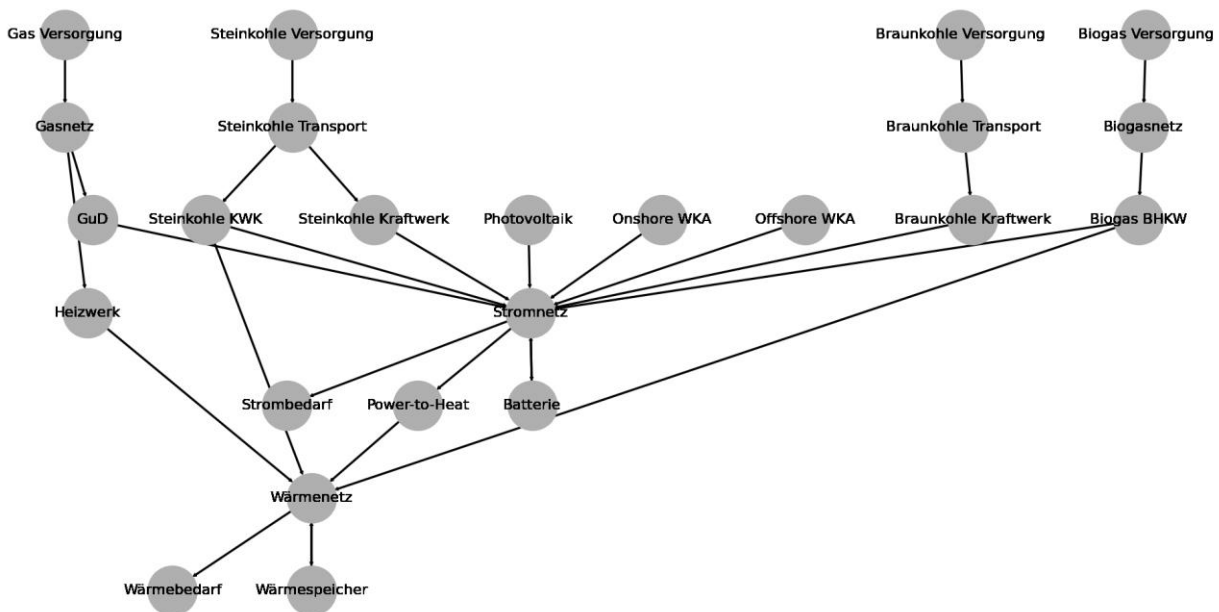


Abbildung 3-6: Visualisierung der Modellarchitektur des *component_es* der *Tessif*-Bibliothek.

Abschließend können die relevanten technischen und ökonomischen Parameter in Tabelle 3-19 und Tabelle 3-20 gefunden werden. Für weitere Informationen bezüglich der Herkunft der Daten und zur Entwicklung der Architektur des ES sei auf die Ausarbeitung von Reimer verwiesen [41].

**Tabelle 3-19: Technische Parameter der Komponenten des *component_es* der *Tessif* Bibliothek.
*entspricht Kapazität in MWh**

Komponente	Inst. Leistung [MW]	Spez. Emissionen [t/MWh]	Wirkungsgrad -
Photovoltaik	1100	0,05	-
Onshore-WKA	1100	0,02	-
Offshore-WKA	150	0,02	-
Biogas-BHKW _{el}	200	0,25	0,40
Biogas-BHKW _{th}	250	0,01875	0,50
BK-KW	500	1	0,40
SK KWK _{el}	300	0,80	0,40
SK KWK _{th}	300	0,06	0,40
SK-KW	500	0,80	0,43
GuD-KW	600	0,35	0,60
Heizwerk	450	0,23	0,90
Power-to-Heat	100	0,0007	0,99
Batterie	100*	0,06	0,90
Wärmespeicher	50*	0	0,85

Tabelle 3-20: Ökonomische Parameter der Komponenten des *component_es* der *Tessif* Bibliothek.

Komponente	Betriebskosten [€/MWh]	Investitionskosten [€/kW]
Photovoltaik	80	1000
Onshore Wind	60	1750
Offshore Wind	105	3900
Biogas-BHKW _{el}	150	3500
Biogas-BHKW _{th}	11,25	262,5
BK-Kraftwerk	65	1900
SK-KWK _{el}	80	1750
SK-KWK _{th}	6	131,25
SK-Kraftwerk	80	1650
GuD-KW	90	950
Heizwerk	35	390
Power-to-Heat	20	100
Batterie	400	1630
Wärmespeicher	20	4,50

4 Auswertung und Vergleich

Bevor die eigentliche Ergebnisauswertung erfolgen kann, muss zunächst ein *Solver* für alle Softwares festgelegt werden, um Vergleichbarkeit zu gewährleisten. In Kapitel 2.4 wird festgestellt, dass der *GLPK*-Solver für komplexere Energiesysteme geeignet ist. Da das ausgewählte Energiesystem darauf abzielt eine vereinfachte Struktur zur Untersuchung einzelner Komponenten und ihres Verhaltens bereitzustellen, ist es ausreichend den *CBC*-Solver zu verwenden. Anfängliche Tests haben darüber hinaus gezeigt, dass nur geringe Unterschiede hinsichtlich der Gesamtergebnisse für das vorliegende Energiesystem entstehen und gleichzeitig die Berechnungszeit durch die Verwendung des *GLPK*-Solver zunimmt. Daher werden alle im Folgenden beschriebenen Optimierungen mit dem *CBC*-Solver durchgeführt.

Begonnen wird damit, die korrekte Integration der in *FINE* möglichen Aggregation der Zeitschritte (Kapitel 2.6.3) und dessen Auswirkung für *Tessif* zu analysieren. Dafür wird das ausgewählte ES mittels der *es2es*-Funktion übersetzt und anschließend mit und ohne Aggregation optimiert. Sowohl die benötigte Berechnungszeit als auch die Ergebnisse durch die *es2mapping*-Funktion sollen verglichen werden, um eine Empfehlung bezüglich der Anwendung in *Tessif* geben zu können.

Die darauffolgende Auswertung beschäftigt sich mit der Durchführung einer Szenarioanalyse des ausgewählten Energiesystems. Währenddessen sollen die drei in *Tessif* integrierten Softwaremodelle hinsichtlich ihrer Ergebnisse analysiert und miteinander verglichen werden. Alle drei Softwaremodelle werden anfangs mittels der Gesamtergebnisse (Kapitel 3.2.2.4) eingeschätzt und anhand der jeweiligen Zusammensetzung der Sektoren Strom und Wärme durch flussbezogene Ergebnisse (Kapitel 3.2.2.3) tiefergehend untersucht. Sofern ein Ausbau möglich ist, sind die erforderlichen installierten Leistungen bzw. Kapazitäten (Kapitel 3.2.2.2) relevant.

Da bereits während der Entwicklung des komponentenbasierten Energiesystems [41] gezeigt werden konnte, dass durch die Softwaremodelle *Oemof* und *PyPSA* sehr ähnliche Ergebnisse berechnet werden, sollten die *FINE*-Ergebnisse in der gleichen Größenordnung liegen. So kann final festgestellt werden, ob die Integration von *FINE* erfolgreich war.

4.1 Aggregation der zeitlichen Auflösung

Wird eine Optimierung mittels *FINE* durchgeführt, kann in *Tessif* die Aggregation der Zeitschritte erfolgen, um die temporale Komplexität und damit die erforderliche Berechnungszeit zu verringern. Da Kotzur et al. [38] ermittelten, dass eine Kostenunterschätzung in Folge einer Spitzenlastglättung entsteht, muss dies auch für die Integration von *FINE* in *Tessif* überprüft werden. Die folgende Tabelle 4-1 enthält die benötigte Zeit, um die Optimierungsaufgabe für das Referenz- und das Basisszenario für das ES aus Abbildung 3-6 mit aggregierter und nicht aggregierter Datengrundlage zu lösen.

Tabelle 4-1: Mittlere Berechnungszeiten der Optimierung durch *FINE* mit und ohne aggregierter Datengrundlage durch *tsam* für die Szenarien (Referenz und Basis) des *component_es* der *Tessif*-Bibliothek.

	Berechnungszeit / min	
	Referenz	Basis
Nicht aggregiert	3,05	15,13
Aggregiert	0,0038	0,20

Die Folge der Aggregation für das Referenzszenario ist, dass lediglich 0,12 % der ursprünglichen Zeit erforderlich ist. Im Basisszenario steigt die Komplexität der Aufgabe, weshalb gleichzeitig auch generell die Rechenzeit verlängert wird. Im aggregierten Zustand erfordert die Optimierung trotzdem für das Basisszenario nur 1,32 % der Zeit gegenüber dem nicht aggregierten. Wie erwartet, kann also eine deutliche Verringerung der in Anspruch genommenen Berechnungszeit mittels der Aggregation durch *tsam* erzielt werden.

In Tabelle 4-2 werden die jeweiligen Gesamtergebnisse zu Emissionen und Kosten beider Optimierungen und Szenarien aufgeführt.

Tabelle 4-2: Gesamtergebnisse zu Kosten und Emissionen optimiert durch *FINE* mit und ohne aggregierter Datengrundlage durch *tsam* für die Szenarien (Referenz und Basis) des *component_es* der *Tessif*-Bibliothek.

	Referenz	Referenz (aggregiert)	Basis	Basis (aggregiert)
Emissionen [kt]	6.815	6.840	250	250
Gesamtkosten [10^3 €]	688.509	686.456	42.289.118	34.680.290
Betriebskosten [10^3 €]	688.509	686.454	734.140	765.618
Investitionskosten [10^3 €]	-	-	41.554.977	33.914.667

Für das Referenzszenario ist zu erkennen, dass mit 25 kt ($< 1\%$) nur eine geringe Abweichung zwischen den berechneten Emissionen entsteht. Die erforderlichen Kosten liegen mit 2,05 Mio. € ($< 1\%$) Unterschied, ebenfalls sehr nahe beieinander. Aufgrund dessen kann für Referenzszenario die Aussage getroffen werden, dass die Aggregation der Zeitschritte nur einen geringen Einfluss ausübt und gleichzeitig eine deutliche Zeitersparnis ermöglicht. Die Emissionen des Basisszenarios fallen aufgrund der übergeordneten Restriktion für beide Varianten gleich aus. Werden dagegen die entstehenden Gesamtkosten betrachtet, so zeigt sich, dass diese mit einer Abweichung von 7,61 Mrd. € ($- 18\%$) den nicht aggregierten Kosten definitiv unterschätzt werden. Während die Betriebskosten ansteigen, werden die Investitionskosten in Folge der Aggregation deutlich gesenkt.

Sowohl die Übereinstimmungen des Referenzszenarios als auch die Unterschiede des Basisszenario sind in Abbildung 4-1 ersichtlich, wenn die Energiemengen in MWh erzeugt durch die jeweiligen Komponenten zusammengestellt werden. Alle zugehörigen absoluten Energiemengen für die beiden untersuchten Sektoren Strom und Wärme befinden sich in Anhang 9 und Anhang 10.

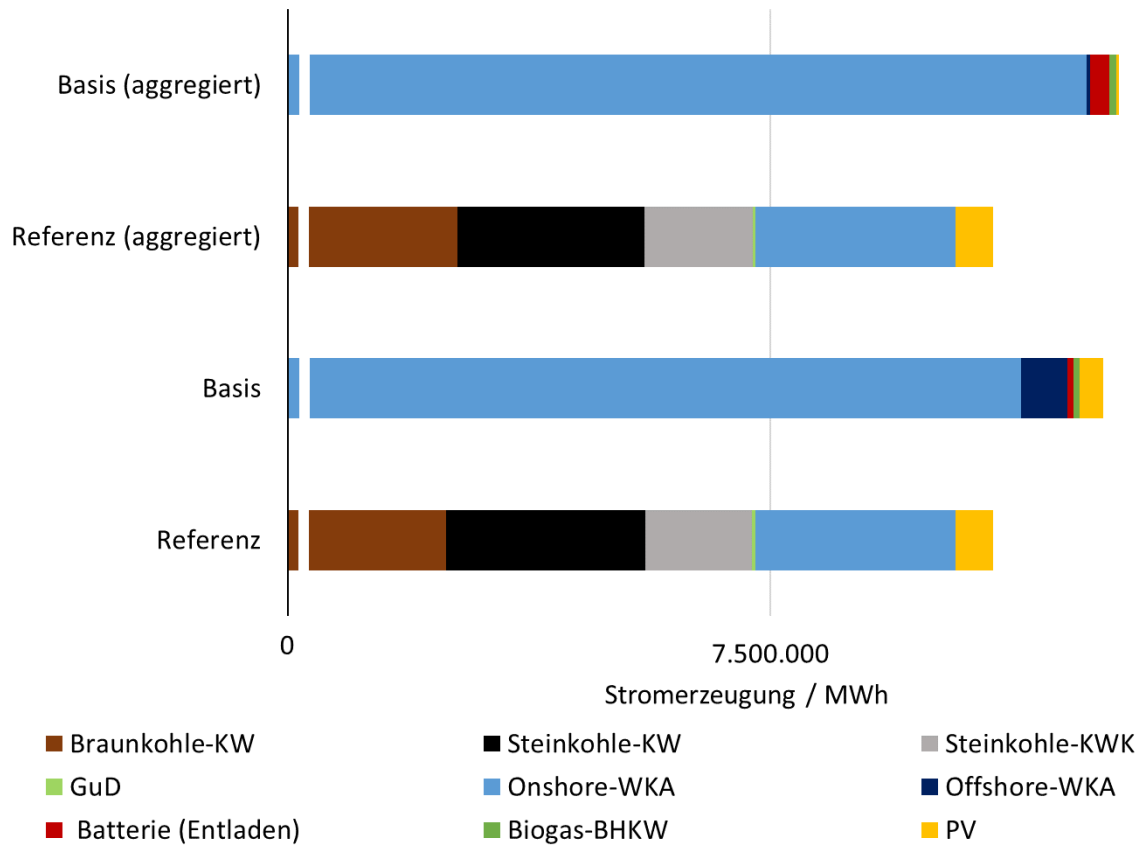


Abbildung 4-1: Zusammensetzung des Stromsektors optimiert durch *FINE* mit und ohne aggregierter Datengrundlage durch *tsam* für die Szenarien (Referenz und Basis) des *component_es* der *Tessif*-Bibliothek.

Analog zu den Gesamtergebnissen sind auch hier die Unterschiede zwischen aggregiertem und nicht aggregiertem Referenzszenario eher gering, wohingegen die Differenzen zwischen den Basisszenarien deutlicher ausfallen. Hier zeigt sich, dass der Anteil an der Strombereitstellung der Offshore-WKA durch die Aggregation verringert wird. Die detaillierten Differenzen im Stromsektor werden in der folgenden Tabelle 4-3 aufgeführt, wobei die angegebenen Werte die relative Abweichung zwischen aggregierten und nicht aggregierten Ergebnissen beschreiben.

Tabelle 4-3: Relative Abweichung der bereitgestellten Strommengen von den Ergebnissen mit aggregierter Datengrundlage durch *tsam* zu denen ohne Aggregation. Optimiert durch *FINE* für die Szenarien (Referenz und Basis) des *component_es* der *Tessif*-Bibliothek.

		SK- KW	BK- KW	GuD	Offshore- WKA	Onshore- WKA	PV	Batterie (Entladen)
Referenz	[GWh]	- 123	+ 117	- 6	-	-	-	-
	[%]	- 6	+ 3	- 18	-	-	-	-
Basis	[GWh]	-	-	- 12	- 441	+ 689	-219	+ 135
	[%]	-	-	-100	- 92	+ 7	- 90	+ 229

Das Biogas-BHKW und die Steinkohle-KWK werden während der Betrachtung des Wärmesektors mit aufgeführt und analysiert. Deutlich wird in Tabelle 4-3, dass der Grundlastherzeuger Braunkohle-KW in Folge der Aggregation 117 GWh (+ 3 %) zusätzlichen Strom bereitstellt. Nahezu identisch ist mit 123 GWh (- 6 %) dagegen die Verringerung der Strommenge, die durch das Steinkohle-KW erzeugt wird. Darüber hinaus sinkt die Menge des GuD-Kraftwerks um 6 GWh (- 18 %). Eine ähnliche Verschiebung der Lasten ist für das Basisszenario ersichtlich. Der Grundlastherzeuger Onshore-WKA erzeugt hier zusätzliche 689 GWh (+ 7 %) Strom, wohingegen die Spitzenlasten durch die Offshore-WKA um 441 GWh (- 92 %) und PV um weitere 219 GWh (- 90 %) verringert werden. Zusammenfassend ist demnach für beide Szenarien ersichtlich, dass die Aggregation eine Verschiebung der Erzeugung von Grund- zur Spitzenlast im Stromsektor bewirkt.

Überprüft werden soll anhand der folgenden Abbildung 4-2, ob diese Verschiebung auch im Wärmesektor auftritt.

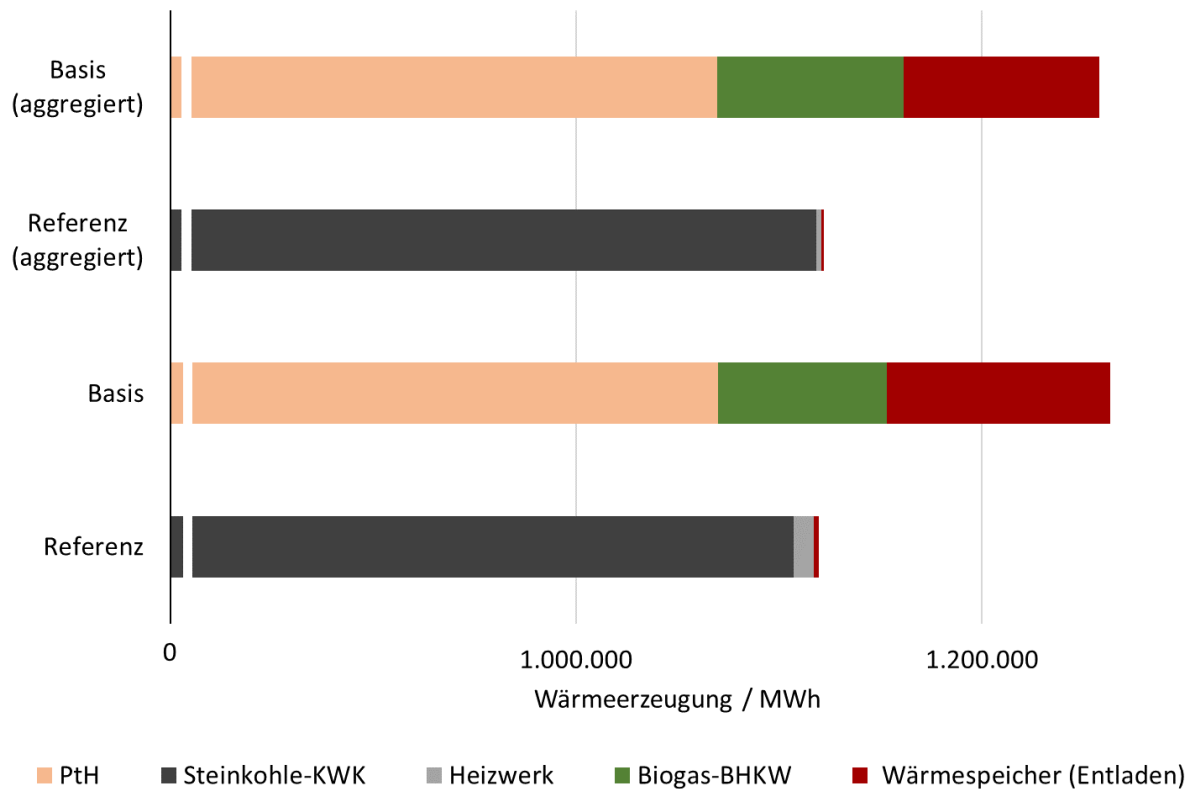


Abbildung 4-2: Zusammensetzung des Wärmesektors optimiert durch *FINE* mit und ohne aggregierter Datengrundlage durch *tsam* für die Szenarien (Referenz und Basis) des *component_es* der *Tessif*-Bibliothek.

Anhand Abbildung 4-2 wird deutlich, dass im Wärmesektor ähnliche, aber weniger deutliche Unterschiede durch die Aggregation für beide Szenarien entstehen. Gründe dafür könnten in einem konstanteren Wärmebedarf der Datengrundlage und in der geringeren Anzahl an Komponenten im Wärmesektor zu finden sein. Durch die detaillierte Betrachtung der Differenzen der bereitgestellten Wärmemengen in Tabelle 4-4, wird das zum Stromsektor analoge Verhalten bestätigt.

Tabelle 4-4: Relative Abweichung der bereitgestellten Wärmemengen von den Ergebnissen mit aggregierter Datengrundlage durch *tsam* zu denen ohne Aggregation. Optimiert durch *FINE* für die Szenarien (Referenz und Basis) des *component_es* der *Tessif*-Bibliothek.

		Biogas-BHKW	SK-KWK	Heizwerk	Wärmespeicher (Entladen)	PtH
Referenz	[GWh]	-	+ 11	- 8	- 1	-
	[%]	-	+ 1	- 77	- 42	-
Basis	[GWh]	+ 9	-	-	- 14	-
	[%]	+ 10	-	-	- 12	-

Für das Referenzszenario stellt die Steinkohle-KWK 11 GWh (+ 1 %) und das Biogas-BHKW 9 GWh (+ 10 %) zusätzliche Wärme bei aggregierter Datengrundlage bereit. Dagegen erzeugt das Heizwerk in Folge der Aggregation 8 GWh (- 77 %) weniger Energie.

Aus Kapitel 2.6.3 geht hervor, dass die Aggregation von Zeitschritten eine Glättung der Spitzenlasten zur Folge hat, wodurch die installierten Leistungen geringer ausfallen müssten und ein konstanteres Profil für die Grundlasterzeuger entstehen würde. Überprüft werden kann dies beispielhaft anhand der installierten Leistungen des Basisszenarios in Tabelle 4-5.

Tabelle 4-5: Installierte Leistungen bzw. Kapazitäten optimiert durch *FINE* mit und ohne aggregierter Datengrundlage durch *tsam* für das Basisszenario des *component_es* der *Tessif*-Bibliothek.

	Nicht aggregiert [MW]	Aggregiert [MW]
Onshore-WKA	15.788	17.665
Offshore-WKA	2.346	150
PV	3.810	1100
GuD	600	600
Biogas-BHKW	516	250
PtH-Anlage	576	311
Heizwerk	450	450
Batterie	2.105	3.106
Wärmespeicher	11.273	1.290

Die installierten Leistungen von GuD-Anlage und Heizwerk wurden in beiden Optimierungen nicht ausgebaut. Eindeutig zu erkennen ist, dass in beiden Sektoren die Spitzenlasten im aggregierten Zustand geringer ausfallen. Dies hat zur Folge, dass im Basisszenario mehr Leistung durch die Grundversorger, beispielsweise Onshore-WKA, bereitgestellt wird und daher die Auslastung konstanter wäre. Überprüft werden kann dies anhand Abbildung 4-3, in der die geordneten Jahresdauerlinien für On- und Offshore-WKA in MW über der Stunde des Jahres aufgetragen werden.

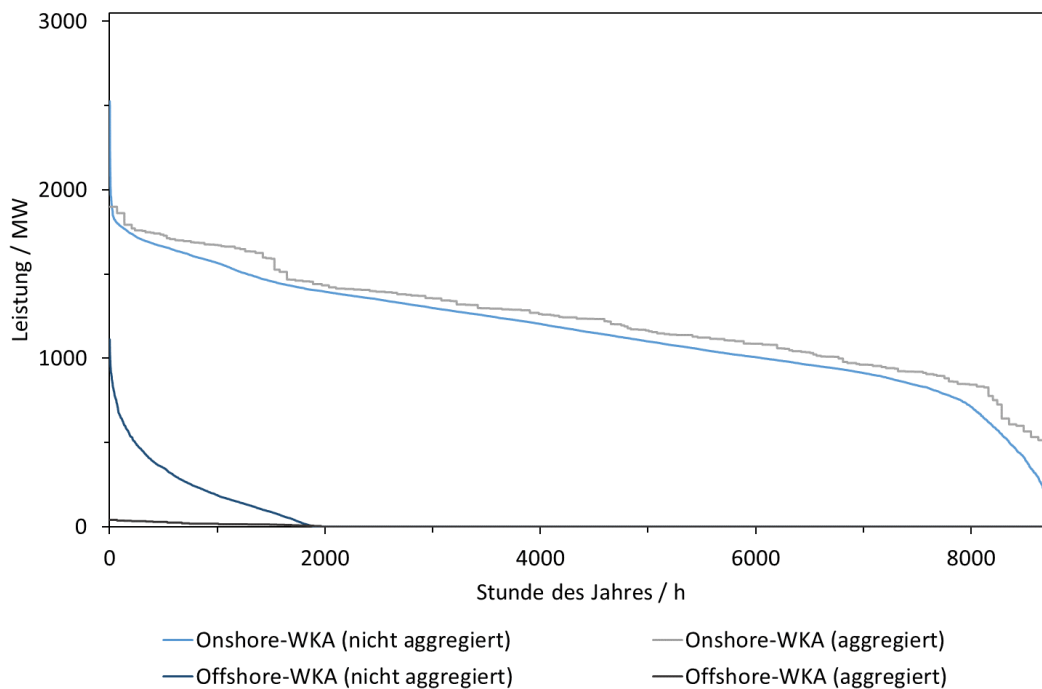


Abbildung 4-3: Jahresdauerlinien der On- und Offshore-WKA optimiert durch *FINE* mit und ohne aggregierter Datengrundlage durch *tsam* für die Szenarien (Referenz und Basis) des *component_es* der *Tessif*-Bibliothek.

Zu erkennen ist, dass die Optimierung mit nicht aggregierten Zeitreihen für die Offshore-WKA wenig Betriebsstunden im Jahr vorsieht, weshalb davon auszugehen ist, dass diese Komponente für die Spitzenlastdeckung eingesetzt wird. Wird dagegen der Verlauf der aggregierten Offshore-WKA betrachtet, zeigt sich, dass diese nicht ausgebaut wurde und ein vernachlässigbar geringer Anteil überhaupt verwendet wird. Der Verlauf der Onshore-WKA ist annähernd waagrecht und deutet somit auf eine konstante Auslastung der Komponente hin, wobei die aggregierte Linie oberhalb der nicht aggregierten liegt. Anhand der Grund- und Spitzenlasterzeuger über die WKA ist somit gezeigt, dass die Mittelwertbildung durch die Aggregation der Zeitreihen zu einer Verschiebung und Vernachlässigung der Extrempunkte führt. Diese Komponenten müssen daher nicht installiert bzw. ausgebaut werden, weshalb eine Verringerung der Investitionskosten (vgl. Tabelle 4-2) die Folge ist.

Damit ist eine erste Einschätzung des Einflusses durch die Aggregation abgeschlossen. Alle weiteren Optimierungen werden im Folgenden ohne die Aggregation der Zeitreihen durchgeführt, sodass eine Vergleichbarkeit der Softwares zur Modellierung und Optimierung von Energiesystemen gewährleistet ist.

4.2 Szenarioanalyse

Das primäre Ziel der Szenarioanalyse dieser Arbeit ist es, die Integration von *FINE* in *Tessif* durch einen systematischen Vergleich zu prüfen. Alle folgenden *FINE*-Optimierungen werden ohne die Aggregation der Zeitreihen durchgeführt, um möglichst gleiche Bedingungen für alle drei Softwares in *Tessif* zu garantieren. Um die Entscheidungsfindung der Softwares nachvollziehen zu können, wird zunächst damit begonnen, Erwartungen an die Ergebnisse der Szenarien des Energiesystems aus Abbildung 3-6 zu formulieren.

Das Referenzszenario zeichnet sich dadurch aus, dass keine Limitierungen bezüglich Emissionen vorgegeben sind, weshalb die Software rein wirtschaftliche Entscheidungen trifft. Weiterhin sollten die entstehenden Gesamtkosten ausschließlich aus Betriebskosten bestehen, da kein Ausbau der Komponenten in diesem Szenario vorgesehen ist. Nach Tabelle 3-20 sind die operationellen Kosten für die die Komponenten eindeutig hierarchisiert, weshalb die Zusammensetzung der Sektoren Strom und Wärme durch die unterschiedlichen Softwaremodelle ähnlich ausfallen sollte. Nur die Steinkohle Kraft-Wärme-Kopplung (SK-KWK), das Steinkohle Kraftwerk (SK-KW) und Photovoltaik (PV) weisen gleiche Kosten auf. Dort ist es wahrscheinlich, dass vorzugsweise die KWK in Anspruch genommen wird, da diese einen erweiterten Nutzen für die Gesamtversorgung ermöglicht. Die Bedingung sollte jedoch sein, dass sowohl Strom als auch Wärme vollständig abgenommen werden. Die durch die Softwaremodelle berechneten Gesamtkosten für das Referenzszenario sollten in der gleichen Größenordnung liegen, wodurch die korrekte Integration von *FINE* für nicht ausbaufähige Szenarien bestätigt werden würde.

Die Erwartung an die Ergebnisse des Basisszenarios ist, dass die Restriktion bezüglich der Emissionen eingehalten wird und infolgedessen die Gesamtkosten für das System höher ausfallen. Damit diese Grenze überhaupt eingehalten werden kann, wird ein Ausbau der Komponenten ermöglicht, weshalb zusätzlich davon auszugehen ist, dass Investitionskosten resultieren. Als wirtschaftlichste und technologisch effizienteste Komponente sollte die Onshore-WKA den Großteil des Strombedarfs erzeugen. Die weiteren erneuerbaren Komponenten werden verwendet, um Spitzenlasten zu decken. Ausschließlich, wenn das Gesamtangebot durch RES zur Versorgung nicht ausreichen sollte, wird auf fossile Erzeu-

gung zurückgegriffen. Von Interesse ist, wie die Software die Versorgung des Wärmesektors gestaltet, da dieser mit SK-KWK und dem Heizwerk (HW) stark emittierende Komponenten beinhaltet. Einzige im Wärmesektor vorhandene erneuerbare Komponente ist das Biogas-BHKW, welches gleichzeitig Strom und Wärme produziert. Sobald überschüssig erneuerbar erzeugter Strom vorhanden ist, sollte dieser durch die Power-to-Heat (PtH)-Anlage zu Wärme umgewandelt werden, um über die Kopplung der Sektoren Emissionen einzusparen. Je nach Angebot und Nachfrage sollte jeglicher Wärmebedarf zunächst durch die PtH-Anlage und ansonsten durch das Biogas-BHKW gedeckt werden. Schlussendlich sollte ausschließlich in Grenzfällen auf fossile Erzeugung zurückgegriffen werden. Analog zu dem Referenzszenario sollten die Gesamtkosten für das Basisszenario in einer ähnlichen Größenordnung liegen, da die technische und wirtschaftliche Hierarchie für die Komponenten eindeutig ist.

4.2.1 Vergleich der Optimierungszeiten

In Kapitel 2.4 wurde herausgearbeitet, dass für Optimierungen ein Kompromiss zwischen Dauer des Prozesses und Detaillierungsgrad der Ergebnisse erforderlich ist. Deshalb wird zunächst der Zeitaufwand der Softwaremodelle verglichen, wobei die Optimierung in *FINE* ohne die Aggregation der Zeitreihen erfolgt. Um eine wissenschaftlich fundierte Aussage treffen zu können, werden beide Szenarien des Energiesystems aus Abbildung 3-6 jeweils dreimal optimiert und die Berechnungszeit jeder Software gemessen. Die Mittelwerte der Zeiten sind in Tabelle 4-6 zu finden, wobei zu erkennen ist, dass weder für die bestehenden Softwares *Oemof* und *PyPSA* noch für *FINE* ein erhöhter Zeitaufwand erforderlich ist.

Tabelle 4-6: Mittlere Berechnungszeiten der Optimierung durch *Oemof*, *PyPSA* und *FINE* für die Szenarien (Referenz und Basis) des *component_es* der *Tessif*-Bibliothek.

	Berechnungszeit / min	
	Referenz	Basis
<i>Oemof</i>	2,68	34,33
<i>PyPSA</i>	1,47	10,43
<i>FINE</i>	3,05	15,13

Die Optimierungszeiten des Referenzszenarios unterscheiden sich im Sekundenbereich, wobei *PyPSA* schneller als *Oemof* und *FINE* ist. Der Unterschied zwischen *FINE* und *PyPSA* ist mit 94,8 Sekunden etwas größer als der zwischen *Oemof* und *PyPSA* mit 22,2 Sekunden. Ähnliche Ergebnisse werden für das Basisszenario erzielt, wo sich erneut *PyPSA* durch Schnelligkeit auszeichnet. *FINE* und *Oemof* benötigen hingegen mehr Zeit für die komplexere Optimierung des Basisszenarios, wobei *Oemof* mit zusätzlichen 19,2 Minuten im Vergleich zu *FINE* die längste Zeit erfordert. *PyPSA* und *FINE* hingegen unterscheiden sich im Basisszenario nur um 4,7 Minuten. Ein Grund für die verkürzte Berechnungszeit von *PyPSA* wurde bereits während der Entwicklung des ES ermittelt. Gezeigt werden konnte, dass eine abweichende Handhabung der spezifischen Emissionen von PtH-Anlage und Batteriespeicher hier verantwortlich für die Verringerung ist [41]. Wird dies berücksichtigt, kann *FINE* im direkten Vergleich zu *Oemof*, basierend auf der gleichen Datengrundlage und Handhabung der Komponenten deutlich Zeit einsparen.

Festzuhalten ist, dass die Optimierungszeit aller drei Softwares in einem akzeptablen Bereich liegt und *FINE* als neu integrierte Software eine vergleichbare Zeit in Anspruch nimmt.

4.2.2 Referenzszenario

Das Referenzszenario des Energiesystems aus Abbildung 3-6 zeichnet sich dadurch aus, dass keine Restriktionen bei der Energieversorgung auferlegt werden und darauf aufbauend das Verhalten des Systems in Folge einer Einsatzplanoptimierung untersucht werden soll. Die erzielten Gesamtergebnisse jeder Software sind in Tabelle 4-7 dargestellt, wobei die Optimierung von *FINE* ohne die Aggregation der Zeitreihen durchgeführt wird.

Tabelle 4-7: Gesamtergebnisse zu Kosten und Emissionen für das Referenzszenario des *componentes* der *Tessif*-Bibliothek optimiert durch *FINE*, *Oemof* und *PyPSA*.

	<i>FINE</i>	<i>Oemof</i>	<i>PyPSA</i>
Emissionen [kt]	6.815	6.814	6.835
Gesamtkosten [10^3 €]	688.509	688.509	688.509
Betriebskosten [10^3 €]	688.509	688.509	688.509
Investitionskosten [10^3 €]	-	-	-

Während die Gesamtkosten von *Oemof*, *PyPSA* und *FINE* übereinstimmen, weisen die Emissionen Differenzen auf. Die Abweichungen ($< 1\%$) zwischen den Softwaremodellen sind jedoch für die Größenordnung vernachlässigbar. Darüber hinaus wird nach Gleichung (3.7) bestätigt, dass die Summe aus Betriebs- und Investitionskosten auf der Grundlage der von *Tessif* generierten Daten identisch mit den Gesamtkosten ist, die während der Optimierungen ermittelt werden.

Erwartungsgemäß liegen auch die Ergebnisse für die Gesamterzeugung der einzelnen Komponenten (vgl. Kapitel 3.2.2.3) sehr nahe beieinander. Sie sind in der folgenden Abbildung 4-4 aufgeführt und die detaillierten Energiemengen jeder Komponente des Stromsektors sind in Anhang 12 zu finden.

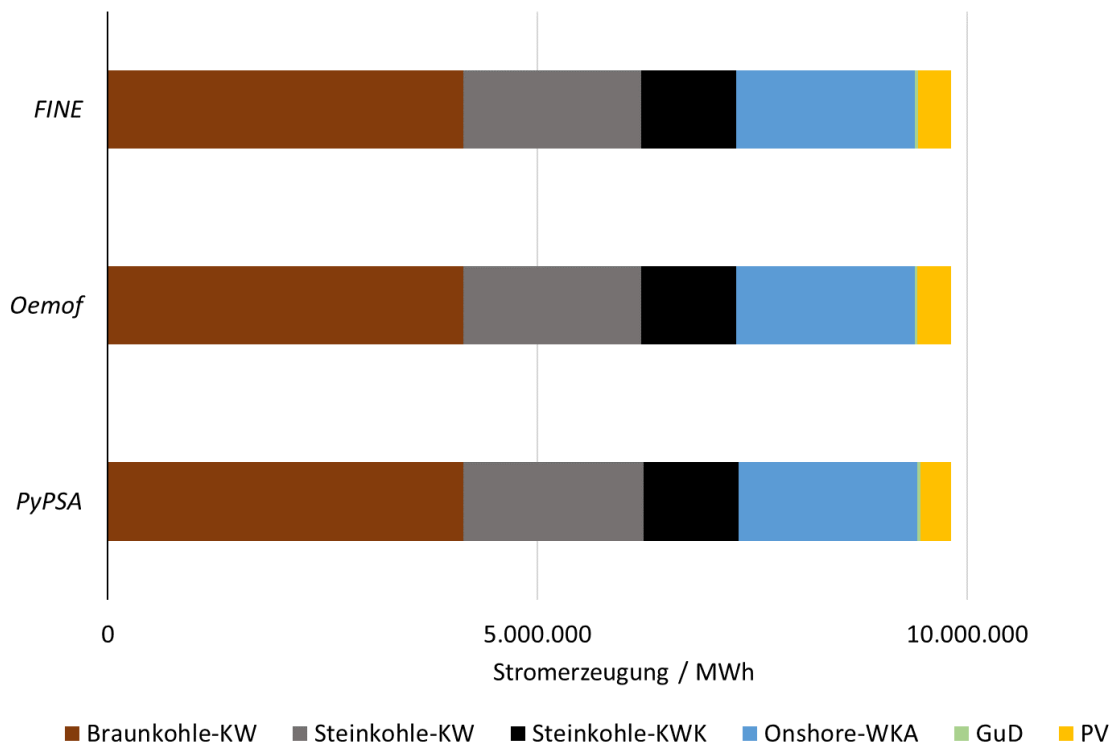


Abbildung 4-4: Zusammensetzung des Stromsektors für das Referenzszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *FINE*, *Oemof* und *PyPSA*.

Braun- (BK) und Steinkohle (SK) liefern den Großteil der benötigten Energie, da sich diese durch eine erhöhte Effizienz bei niedrigen Kosten ausweisen. Zusätzlich ermöglicht die SK-KWK Anlage gleichzeitig Strom und Wärme zur Verfügung zu stellen, weshalb die Entscheidung der Software zur Nutzung dieser nachvollzogen werden kann. Ein geringer, aber dennoch vorhandener Anteil der Residuallast beinhaltet das GuD-Kraftwerk, welches bei jeder Software für Spitzenlasten Verwendung findet. Der übrige Teil wird durch die vorhandenen erneuerbaren Energien Onshore-WKA und PV gedeckt. Um mögliche geringfügige Unterschiede zu erkennen, werden die Abweichungen der von den drei Softwares berechneten gesamten Leistungen in der nachstehenden Tabelle 4-8 aufgeführt. Die Werte sind bezogen auf die *FINE*-Ergebnisse und beschreiben die Abweichung der zur Verfügung gestellten Energie in MWh durch die jeweilige Komponente von *Oemof* und *PyPSA* zu *FINE*.

Tabelle 4-8: Abweichung der bereitgestellten Strommengen von *Oemof* und *PyPSA* relativ zu *FINE* für das Referenzszenario des *component_es* der *Tessif*-Bibliothek.

		GuD	SK-KWK	SK-KW	BK-KW	Onshore-WKA	PV
<i>Oemof</i>	[MWh]	-	- 20	- 2.359	+ 26	- 5	+ 2.359
	[%]	-	< + 1	< - 1	< + 1	< - 1	< + 1
<i>PyPSA</i>	[MWh]	-	- 20	+ 26.795	+ 26	- 5	- 26.795
	[%]	-	< - 1	+ 1	< + 1	< - 1	- 7

Die weiteren im Energiesystem (vgl. Abbildung 3-6) vorhandenen Komponenten Biogas-BHKW, Offshore-WKA, Batteriespeicher und PtH-Anlage werden aufgrund der wirtschaftlichen Nachteile (vgl. Tabelle 3-20) durch keine Software verwendet und daher nicht mit aufgeführt. Auffällig ist die in *PyPSA* um 26.795 MWh (- 7 %) gegenüber *FINE* verringerte PV-Energiemenge, welche der durch das SK-Kraftwerk zusätzlich erbrachten Menge entspricht. Demgegenüber zeigt *Oemof* eine Verschiebung der bereitgestellten Energie um 2.359 MWh (+ 1 %) zur PV, verringert bei dem SK-Kraftwerk. Da die Betriebskosten der beiden Komponenten übereinstimmen (vgl. Tabelle 3-20), entscheidet *PyPSA* Energie verstärkt mittels des SK-Kraftwerks bereitzustellen, wohingegen *FINE* und *Oemof* die PV-Komponente verwenden. *Oemof* kann im Vergleich die größte Energiemenge mittels PV bereitstellen, welche sich dennoch nur geringfügig von der gesamten PV-Energie durch *FINE* unterscheidet. Daher ist festzustellen, dass im Stromsektor Ähnlichkeiten zwischen den Modellen und besonders zwischen *FINE* und *Oemof* aufgezeigt werden können.

Mit Hilfe der in Kapitel 3.2.2.1 vorgestellten Methode zur Erfassung der ermittelten Daten kann das Erzeugungsprofil erstellt werden. Zwischen den Softwaremodellen zeigen sich auch hier Übereinstimmungen sowohl für einen Sommer- als auch für einen Wintermonat. Aufgetragen ist in Abbildung 4-5 die zur Verfügung gestellte Leistung aller Komponenten über einen Wintermonat (15. Januar – 15. Februar) optimiert durch *FINE*.

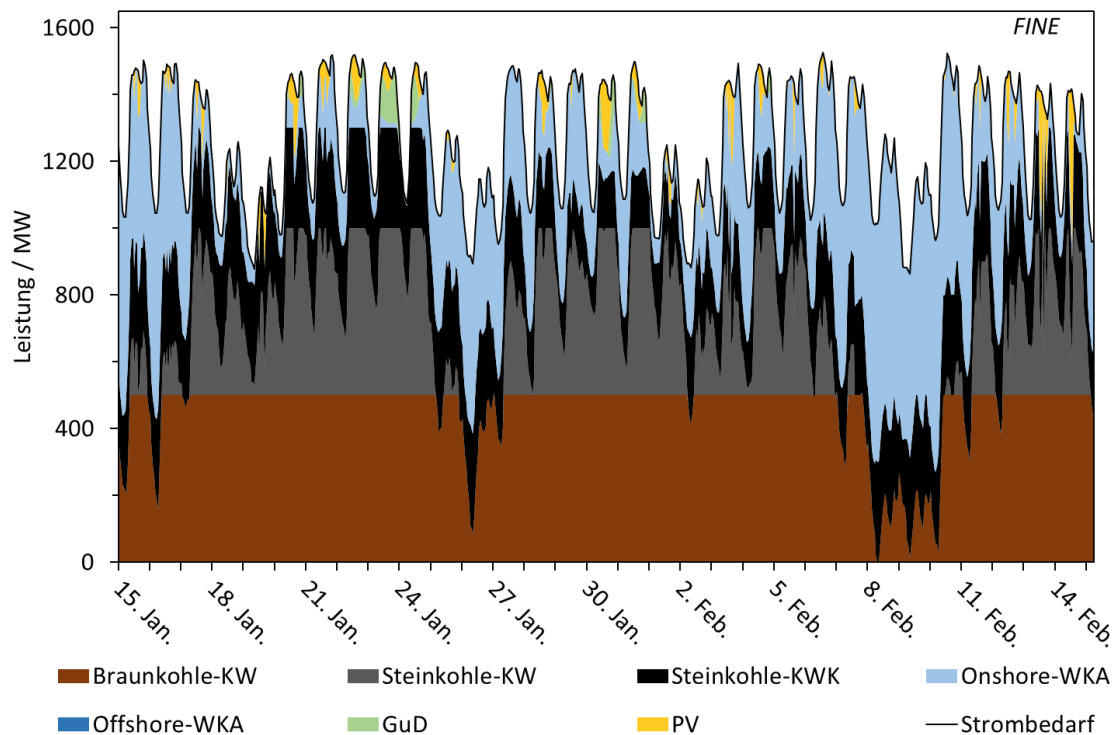


Abbildung 4-5: Erzeugungprofil des Stromsektors eines Wintermonats für das Referenzszenario des *component*es der *Tessif*-Bibliothek optimiert durch *FINE*.

Die durch *Oemof* und *PyPSA* sehr ähnlichen Profile für den Stromsektor können Anhang 1 entnommen werden. Zu erkennen ist, dass die Grundlast mittels der fossilen Energieerzeuger gedeckt wird, bei gleichzeitig voller Ausschöpfung der im Vergleich günstigen Stromerzeugung durch Onshore-WKA und PV. Die fossile Erzeugung über BK zeichnet sich durch niedrig ausfallende Betriebskosten aus und bedient daher den größten Anteil der Residuallast. Die Erzeugung durch das SK-KW und SK-KWK weist identische Betriebskosten auf, weshalb durch die Software entschieden werden muss, welche die effizientere Wahl darstellt. Bedingung für die gleichzeitige Erzeugung von Strom und Wärme ist, dass beide Energieträger abgenommen werden müssen und keine Überproduktion vorliegt. Je nach Wärmebedarf entscheidet *FINE* daher im Winter primär Energie gekoppelt zu erzeugen, um die Kosten zu minimieren. Der nicht durch BK, SK oder RES zu deckende Anteil des Stroms wird als Spitzenlast über das GuD-Kraftwerk bereitgestellt.

Besonders Interessant hinsichtlich des Erzeugungsverhaltens in Abbildung 4-5, sind die Wochenenden (25. – 27. Januar und 8. – 11. Februar). Deutlich wird, dass in Folge eines verringerten Strombedarfs zuerst das SK-Kraftwerk abgefahren wird. Für einen kurzen

Zeitpunkt des 8. Februar kann die Residuallast auf die Energiebereitstellung durch SK-KWK begrenzt werden. Dieser Zeitpunkt verdeutlicht, dass diese Alternative die wirtschaftlichste der fossilen Erzeuger ist. Effizienter sind demnach ausschließlich die Onshore-WKA und die PV, wobei aufgrund der Uhrzeit sowie Jahreszeit keine Sonne vorhanden ist und der RES Strom ausschließlich über Windkraft erzeugt wird.

Die folgende Abbildung 4-6 zeigt das entstandene Erzeugungsprofil, aufgetragen in MW über einen Sommermonat (14. Juli – 14. August).

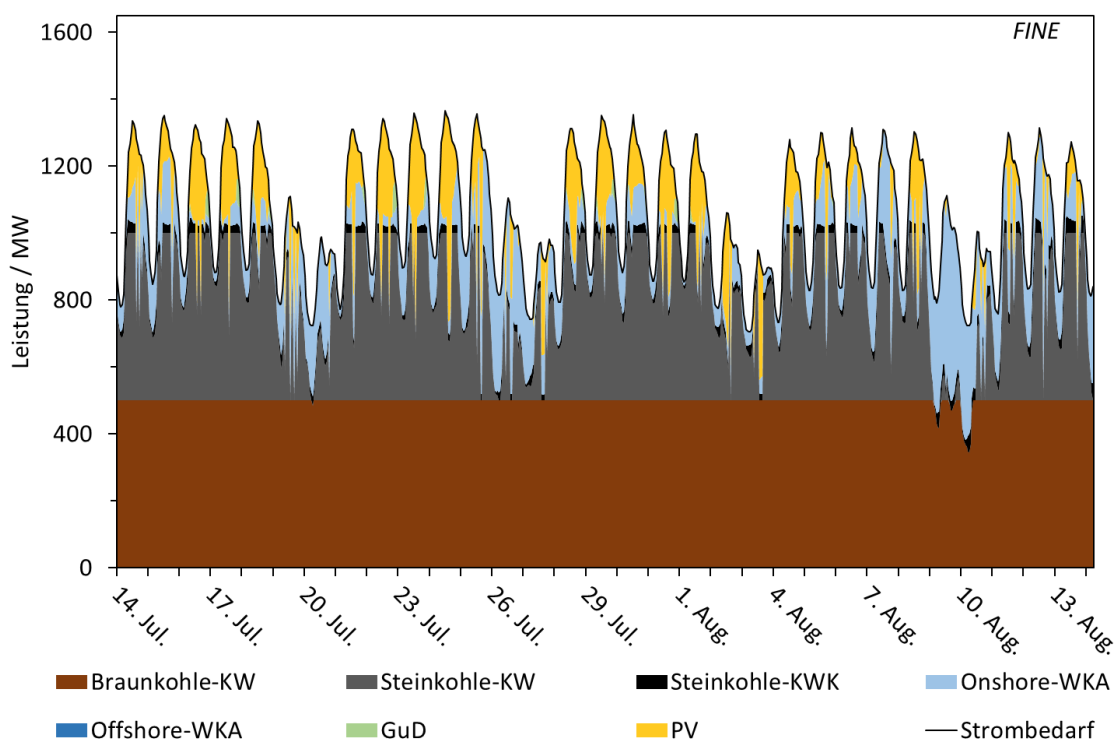


Abbildung 4-6: Erzeugungsprofil des Stromsektors eines Sommermonats für das Referenzszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *FINE*.

Die zur Verfügung stehende Leistung durch PV ist saisonbedingt angestiegen, während gleichzeitig die Anteile des GuD-Kraftwerks und der SK-KWK nahezu nicht vorhanden sind. Die Begründung dafür liegt in der verringerten Nachfrage und den gleichzeitig höheren Betriebskosten, weshalb insgesamt weniger Leistung durch diese Komponenten erforderlich ist. Die sehr ähnlichen Erzeugungsprofile von *Oemof* und *PyPSA* können Anhang 2 entnommen werden.

Die gleichzeitige Bereitstellung von Strom und Wärme soll Aufschluss darüber geben, wie das jeweilige Softwaremodell mit der sektoralen Kopplung umgeht. Daher wird der Wärmesektor detaillierter betrachtet, um mögliche Unterschiede in der Verwendung zu ermitteln. Abbildung 4-7 stellt die gesamte Wärmeerzeugung nach Anhang 13 jeder Komponente und Software für das Referenzszenario zusammen.

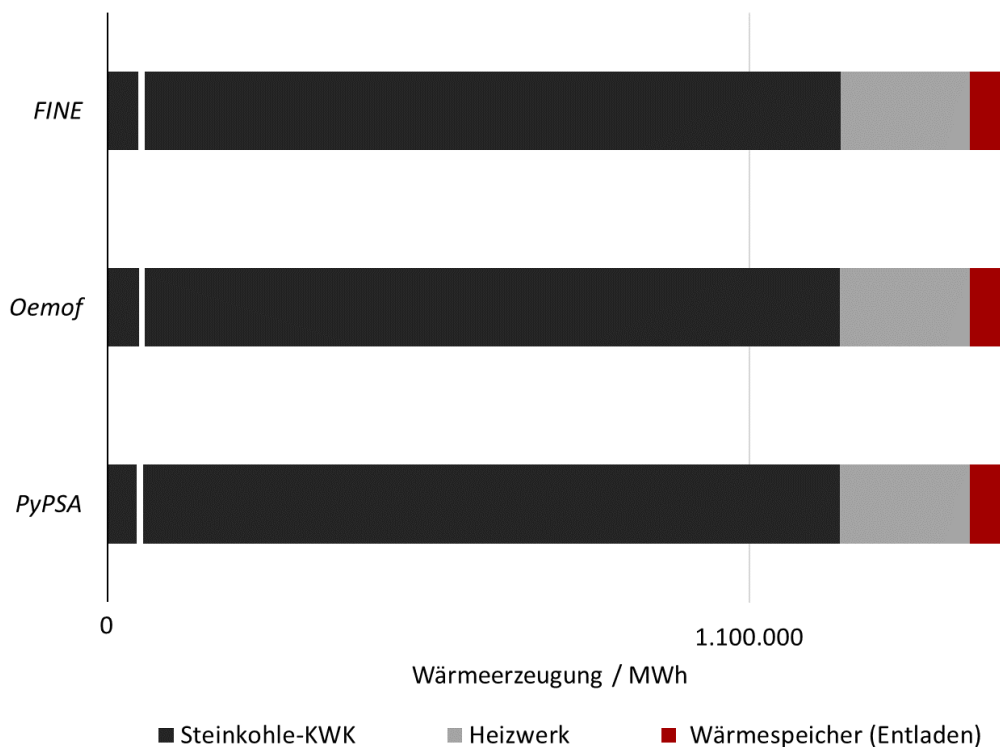


Abbildung 4-7: Zusammensetzung des Wärmesektors für das Referenzszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *FINE*, *Oemof* und *PyPSA*.

Zu erkennen ist, dass der Großteil der Wärme mittels der SK-KWK und Spitzenlasten über das Heizwerk sowie den Wärmespeicher bereitgestellt werden. Da das Energiesystem generell eine begrenzte Auswahl an Komponenten für die Wärmebereitstellung beinhaltet, werden die Komponenten durch jede Software annähernd identisch beansprucht.

Wird das Verhalten aller Komponenten des Wärmesektors für einen Wintermonat (15. Januar – 15. Februar) in Abbildung 4-8 betrachtet, bestätigt sich die eingangs formulierte Erwartung an die Grund- und Spitzenlastversorgung.

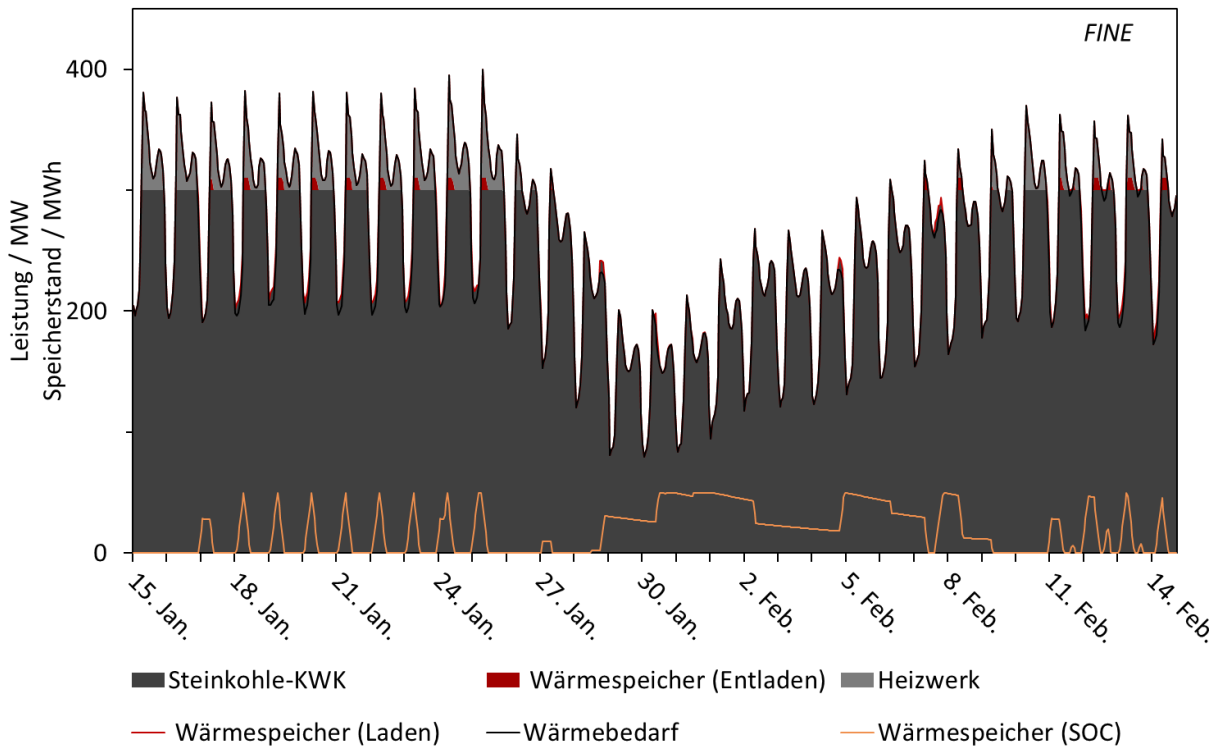


Abbildung 4-8: Erzeugungprofil des Wärmesektors eines Wintermonats für das Referenzszenario des *component_es* der *Tessif-Bibliothek* optimiert durch *FINE*.

Die Erzeugungprofile für *Oemof* und *PyPSA* können Anhang 3 entnommen werden. Eingeteilt werden kann Abbildung 4-8 in zwei Perioden mit unterschiedlichem Verhalten. Am Anfang der betrachteten Zeitspanne (15. – 27. Januar) kann eine wiederkehrende Spitzenlast beobachtet werden, die sowohl durch den Wärmespeicher als auch durch das Heizwerk bedient wird. Fällt der Bedarf kurzzeitig aufgrund des Tag- und Nachtrhythmus, kann überschüssige Energie im Wärmespeicher zwischengelagert werden. Sobald der Bedarf wieder steigt, wird die gespeicherte Energie verwendet, um die im Verhältnis teure Erzeugung des Heizwerks zu minimieren. Die zweite Phase beschreibt geringere Nachfrage (28. Januar – 10. Februar) unterhalb der installierten Leistung der SK-KWK. Es wird für diesen Fall auf die im Speicher vorhandene Energie verzichtet. Grund dafür sind die wirtschaftlichen Vorteile der kombinierten Erzeugung von Strom und Wärme, welche die Vorteile des Wärmespeichers überwiegen.

Anhand dessen lässt sich ableiten, dass, wenn Strom und Wärme gleichzeitig benötigt werden, diese priorisiert mittels der KWK-Komponenten bereitgestellt werden. Spitzenlasten, die diese Leistung übersteigen, werden zuerst durch den Wärmespeicher und dann durch das Heizwerk bedient. Diese Abstufung der wirtschaftlichen und gleichzeitig effizienten Versorgung wird durch die weiteren Softwaremodelle *Oemof* und *PyPSA* identisch getroffen. Die jeweiligen Erzeugungsprofile können dem Anhang entnommen werden.

Tabelle 4-9 führt die Abweichungen der Energiebereitstellung durch Komponenten im Wärmesektor auf, welche von *Oemof* und *PyPSA* gegenüber *FINE* erneut jeweils prozentual unter 10 % liegen. Die enthaltenen Werte beschreiben die Differenz zwischen den bereitgestellten Energiemengen der Komponenten von *Oemof* und *PyPSA* gegenüber *FINE*.

Tabelle 4-9: Abweichung der bereitgestellten Wärmemengen von *Oemof* und *PyPSA* relativ zu *FINE* für das Referenzszenario des *component_es* der *Tessif*-Bibliothek.

		SK-KWK	Heizwerk	Wärmespeicher (Entladen)	Wärmespeicher (Laden)
<i>Oemof</i>	[MWh]	- 20,44	-	+ 20,44	-
	[%]	< - 0,10	-	< + 0,10	-
<i>PyPSA</i>	[MWh]	- 20,44	-	+ 20,44	-
	[%]	< - 0,10	-	< + 0,10	-

Die PtH-Anlage und das Biogas-BHKW finden aufgrund von wirtschaftlichen Nachteilen für den Wärmesektor in keinem der Softwaremodelle Anwendung. *Oemof* und *PyPSA* benötigen 20,44 MWh (< - 0,10 %) weniger Wärme durch die SK-KWK und können diesen Anteil über die Wärmespeicher bereitstellen. *FINE* kann demnach bei gleicher Ladeleistung die in den Speicher fließt, weniger Energie entnehmen, um den Wärmebedarf zu decken. Relevante technische Parameter des Wärmespeichers für entstehende Verluste werden durch Ein- und Ausspeicherwirkungsgrade von jeweils 0,95 % sowie durch eine Selbstentladung von 0,5 % pro Zeitschritt beschrieben. Der Gesamtwirkungsgrad des Wärmespeichers

$$\eta_{WS}^{FINE} = \frac{E_{W,Entladen}^{FINE}}{E_{W,Laden}^{FINE}} = \frac{2.427,70 \text{ MWh}}{3.478,32 \text{ MWh}} = 0,6979 = 69,79 \% \quad (4.1)$$

in *FINE* wird bestimmt, indem aus Anhang 14 die entladene Energiemenge $E_{W,Entladen}$ durch die geladene Energiemengen $E_{W,Laden}$ geteilt wird. Die Speicherdissipation berechnet sich aus der Differenz des Wirkungsgrades η_{WS}^{FINE} und vollen 100 %. Sie beträgt demnach in *FINE* für den Wärmespeicher 30,21 %. Werte für *Oemof* und *PyPSA* werden analog berechnet und alle Ergebnisse können Tabelle 4-10 entnommen werden.

Tabelle 4-10: Speicherdissipation durch Selbstentladung in Folge der Optimierung durch *FINE*, *Oemof* und *PyPSA* für das Referenzszenario des *component_es* der *Tessif*-Bibliothek.

Wärmespeicherdissipation	
<i>PyPSA</i>	29,63 %
<i>Oemof</i>	29,63 %
<i>FINE</i>	30,21 %

Da jeder Software der gleiche Speicherwirkungsgrad für die Optimierung vorgegeben wird, können die Abweichungen ausschließlich auf die Standverluste zurückgeführt werden. Die Differenz der Dissipationen von *FINE* zu *Oemof* und *PyPSA* im Wärmesektor von 0,59 % kann anschließend genutzt werden, um die über den Wärmespeicher eingesparte Energiemenge

$$\Delta E_{W,FINE} = 0,59 \% \cdot 3.478,32 \text{ MWh} = 20,44 \text{ MWh} \quad (4.2)$$

zu berechnen. Hierfür wird die prozentuale Abweichung mit der geladenen Energiemenge des Wärmespeichers $E_{WS,Laden}^{FINE}$ multipliziert, wobei das Ergebnis den in Tabelle 4-9 weniger umgesetzten 20,44 MWh Wärme entspricht.

Ein möglicher Grund für den erhöhten Verlust sind die Speicherfüllstände zu Beginn und am Ende der Zeitspanne. In Tabelle 4-11 werden diese für die jeweilige Software aufgeführt, wobei auffällt, dass *PyPSA* und *Oemof* im Gegensatz zu *FINE* in einem neutralen Zustand beginnen.

Tabelle 4-11: Wärmespeicherstand für den ersten und letzten Zeitschritt in Folge der Optimierung durch *FINE*, *Oemof* und *PyPSA* für das Referenzszenario des *component_es* der *Tessif*-Bibliothek.

Zeitschritt	<i>FINE</i> [MWh]	<i>PyPSA</i> [MWh]	<i>Oemof</i> [MWh]
1	5,79	0	0
⋮	⋮	⋮	⋮
8760	5,82	0	0

Deutlich wird der Zeitpunkt der Dissipation in Abbildung 4-9, wenn der Speicherstand in MWh für das Ende der untersuchten Zeitspanne aufgetragen wird.

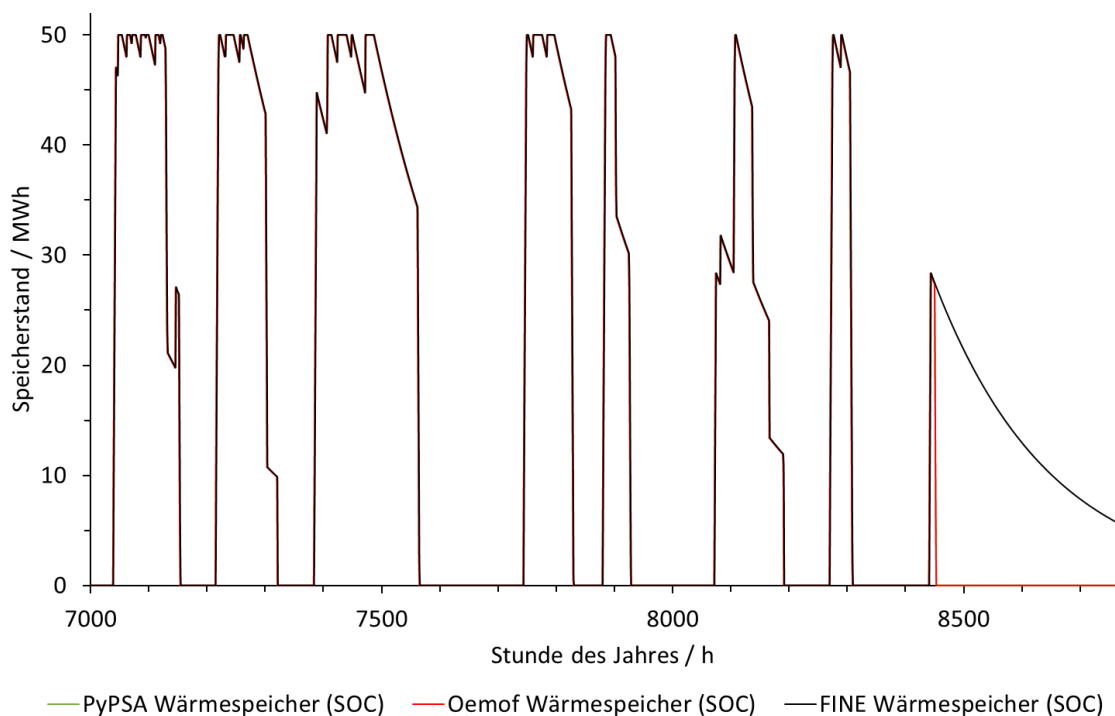


Abbildung 4-9: Ladezustand des Wärmespeichers für das Referenzszenario des *component_es* der *Tessif*-Bibliothek für das Ende der untersuchten Zeitperiode optimiert durch *FINE*, *Oemof* und *PyPSA*.

Während die drei Softwaremodelle die Speicherstände für den Großteil der untersuchten Zeitspanne deckungsgleich auslegen, zeigt sich gegen Ende ein deutlicher Unterschied. *PyPSA* und *Oemof* folgen weiterhin einem identischen Verlauf und entladen jegliche Energie bis zum 19. Dezember um 03:00 Uhr (Stunde des Jahres: 8452 h). *FINE* dagegen hält diesen Füllstand und verwendet bis zum Ende der Zeitspanne den Speicher nicht weiter, weshalb 21,57 MWh aufgrund der definierten Selbstentladung entweichen. Der finale

Speicherstand mit 5,82 MWh (11,64 % der Kapazität) abzüglich der Selbstentladung entspricht dann dem Füllstand zu Beginn der Zeitperiode. Der Speicherstand für den ersten Zeitschritt wird durch *Tessif* für *Oemof* und *PyPSA* standardmäßig mit 0 MWh vorgegeben, wohingegen über die Integration von *FINE* kein Einfluss möglich ist und die Optimierung eine Anpassung vornimmt. Um ein aussagekräftiges Ergebnis bezüglich der Gesamtbilanz zu ermitteln, wählt *FINE* den finalen Speicherstand, sodass dieser den Initialen um die Selbstentladerate von 0,5 % übersteigt. Dadurch wird verhindert, dass Vorteile anhand eines hohen anfänglichen Speicherstandes in der Gesamtbilanz entstehen. Sobald in *Tessif* ein periodischer Speicher definiert wird, indem der anfängliche Füllstand gleich dem am Ende gesetzt wird, ist davon auszugehen, dass die anderen Softwares ein ähnliches Verhalten zeigen.

Sowohl die Gesamtergebnisse der Kosten und Emissionen als auch die Verwendung der zur Verfügung stehenden Komponenten sind auf Grundlage dieser Auswertung für die drei untersuchten Softwaremodelle nahezu identisch. Für das Referenzszenario zeigen sich deutliche Parallelen zwischen den Softwaremodellen, weshalb für die enthaltenen Komponenten und Parameter auf eine korrekte Integration von *FINE* und *Tessif* geschlossen werden kann.

4.2.3 Basisszenario

Das Basisszenario des Energiesystems aus Abbildung 3-6 sieht die Möglichkeit vor, die installierte Leistung zu erweitern, um die festgelegte Beschränkung von 250.000 t Emissionen einzuhalten. Tabelle 4-12 enthält die erzielten Gesamtergebnisse der drei Softwareremodelle, wobei die Optimierung in *FINE* erneut ohne Aggregation der Zeitreihen erfolgt.

Tabelle 4-12: Gesamtergebnisse zu Kosten und Emissionen für das Basisszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *FINE*, *Oemof* und *PyPSA*.

	<i>FINE</i>	<i>Oemof</i>	<i>PyPSA</i>
Emissionen [kt]	250	250	265
Gesamtkosten [10^3 €]	42.289.118	42.289.118	37.793.513
Betriebskosten [10^3 €]	734.140	734.140	820.206
Investitionskosten [10^3 €]	41.554.977	41.554.978	36.973.307

Während *FINE* und *Oemof* die Grenze einhalten, übersteigen die durch *PyPSA* ermittelten Emissionen die Restriktion um 15 kt (+ 6 %) und es resultiert eine Gesamtkostenersparnis von 4,5 Mrd. € (- 11 %). Zwischen *Oemof* und *FINE* unterscheiden sich die berechneten Kosten nicht, weshalb hier eine erste Ähnlichkeit im Basisszenario erkannt werden kann. Das vorliegende komponentenbasierte Energiesystem wurde bereits mittels *Tessif* untersucht. Gezeigt werden konnte, dass sich Gesamtkosten angleichen, wenn die spezifischen Emissionen von Batteriespeicher und PtH-Anlage vernachlässigt werden [41].

Erneut geprüft werden kann die Vernachlässigung in *PyPSA* mit Hilfe der in dieser Arbeit ermittelten Energiemengen für die Komponenten (vgl. Anhang 12, Anhang 13) multipliziert mit den jeweiligen spezifischen Emissionen (vgl. Tabelle 3-19). Somit ergeben sich die Emissionen für die Batterie

$$e_{\text{Batterie}}^{\text{PyPSA}} = 237.491,70 \text{ MWh} \cdot 0,06 \frac{\text{t}}{\text{MWh}} = 14.249,5 \text{ t} \quad (4.3)$$

und für die PtH-Anlage

$$e_{\text{PtH}}^{\text{PyPSA}} = 1.064.803,78 \text{ MWh} \cdot 0,0007 \frac{\text{t}}{\text{MWh}} = 745,36 \text{ t} \quad (4.4)$$

Die Summe aus Batterie und PtH-Anlage entspricht den zusätzlich berechneten Emissionen aus Tabelle 4-12 für *PyPSA*, welche durch *Tessif* im Nachgang der Optimierung für die Vergleichbarkeit addiert wurden. Durch eine Sensitivitätsanalyse am Ende dieses Kapitels soll für *FINE* gezeigt werden, dass eine Annäherung der Gesamtergebnisse zwischen den Softwares entsteht, sobald die spezifischen Emissionen von Batterie und PtH-Anlage entfernt werden.

Die über Gleichung (3.7) in *Tessif* berechneten Betriebs- und Investitionskosten aus Tabelle 4-12 zeigen für alle drei Softwares Unterschiede zu den Gesamtkosten, die jeweils in einem hinreichend genauen Bereich liegen. Werden diese detailliert betrachtet, fällt auf, dass die Investitionskosten in *PyPSA* im Vergleich zu *Oemof* und *FINE* gesenkt werden können, wohingegen die Betriebskosten ansteigen.

Investitionskosten entstehen, wenn Komponenten durch die jeweilige Software ausgebaut werden, um die vorgegebenen Restriktionen einzuhalten. Die *es2mapping*-Funktion ermöglicht es, die installierten Leistungen bzw. Kapazitäten zusammenzustellen (vgl. Kapitel 3.2.2.2). Sie sind in der folgenden Tabelle 4-13 aufgeführt.

Tabelle 4-13: Installierte Leistungen bzw. Kapazitäten optimiert durch *FINE*, *Oemof* und *PyPSA* für das Basisszenario des *component_es* der *Tessif*-Bibliothek. Relative Abweichung zu *FINE*

	<i>FINE</i>	<i>Oemof</i>	<i>PyPSA</i>	Rel. Abweichung <i>PyPSA</i> zu <i>FINE</i>
	[MW]	[MW]	[MW]	[%]
Onshore-WKA	15.788	15.788	12.788	- 19
Offshore-WKA	2.346	2.346	1.961	- 16
PV	3.810	3.810	3.695	- 3
GuD-KW	600	600	600	0
Batterie	2.105	2.105	3.685	+ 75
Biogas-BHKW	516	516	549	+ 6
PtH-Anlage	576	576	621	+ 8
Heizwerk	450	450	450	0
Wärmespeicher	11.273	11.273	11.014	- 2

FINE und *Oemof* legen alle Komponenten identisch aus, wohingegen *PyPSA* für die Grundlastversorgung durch Onshore-WKA (- 19 %) und Offshore-WKA (- 16 %) deutlich weniger zusätzliche Leistung installiert. Die Komponenten Batteriespeicher (+ 75 %), PtH-Anlage (+ 8 %) sowie das Biogas-BHKW (+ 6 %) weisen dagegen in *PyPSA* eine höhere Leistung bzw. Kapazität auf. Geringer fallen demgegenüber die Abweichungen der Komponenten PV (- 3 %), und Wärmespeicher (- 2 %) von *PyPSA* zu *FINE* aus. Die einzige Übereinstimmung zwischen allen drei Softwares sind die installierten Leistung von GuD-Kraftwerk und Heizwerk, die jeweils nicht ausgebaut werden. Der Grund dafür liegt erneut in den zugehörigen technischen und ökonomischen Parametern, die darauf hindeuten, dass diese Komponenten ausschließlich für Spitzenlasten Anwendung finden. Eine Überprüfung kann anhand der Verteilung der insgesamt bereitgestellten Strommenge durch alle Komponenten in Abbildung 4-10 erfolgen. Die zugrundeliegenden detaillierten Energiemengen befinden sich in Anhang 12.

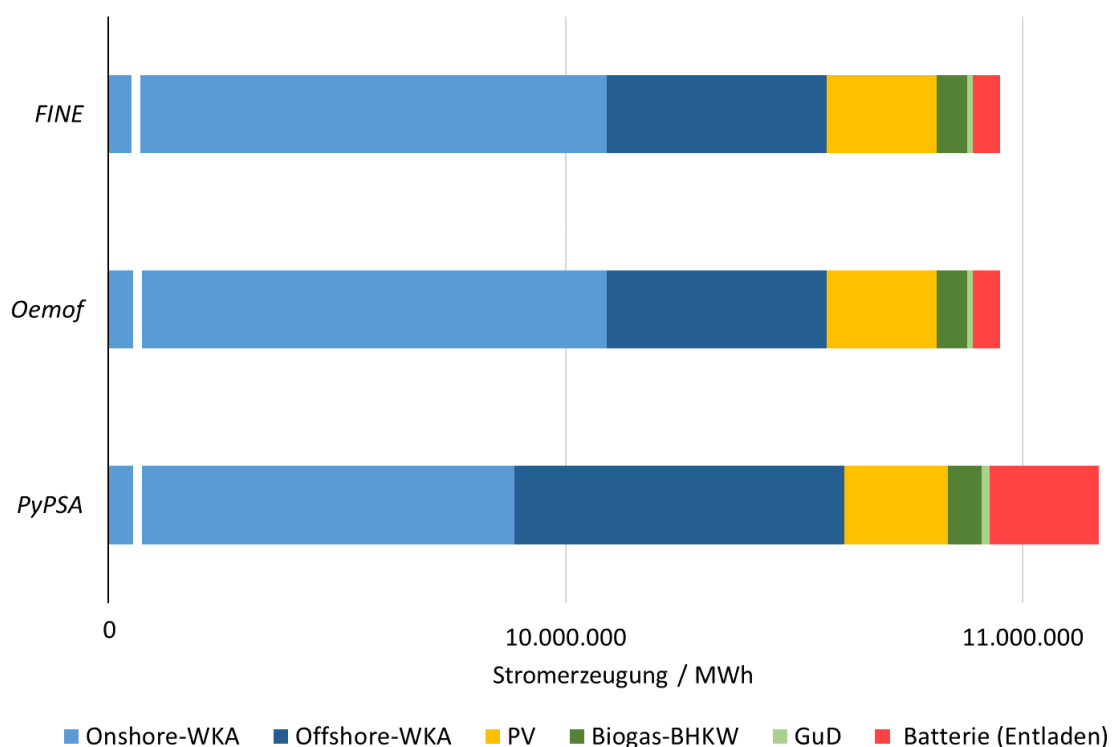


Abbildung 4-10: Zusammensetzung des Stromsektors für das Basisszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *FINE*, *Oemof* und *PyPSA*.

Wie erwartet, wird die Grundlast durch jede Software mittels Onshore-WKA gedeckt und die geringeren Anteile werden über Offshore-WKA, PV und GuD-Anlage bereitgestellt. Die

Unterschiede von *PyPSA* zu den beiden anderen Softwares werden deutlich und die Abweichung der Gesamtkosten aus Tabelle 4-12 kann auf die Komponenten Offshore-WKA und Batteriespeicher zurückgeführt werden. *Oemof* und *FINE* hingegen weisen in dieser Darstellung erneut Parallelen auf. Werden diese Abweichungen des Stromsektors in Tabelle 4-14 tiefergehend betrachtet, zeigt sich, dass *FINE* und *Oemof* das Energiesystem deckungsgleich versorgen und *PyPSA* zusätzlich 216 GWh (+ 2 %) Strom bereitstellt.

Tabelle 4-14: Abweichung der bereitgestellten Strommengen von *Oemof* und *PyPSA* relativ zu *FINE* für das Basisszenario des *component_es* der *Tessif*-Bibliothek.

		Biogas-BHKW	GuD	Offshore-WKA	Onshore-WKA	PV	Batterie (Entladen)
<i>Oemof</i>	[GWh]	-	-	-	-	-	-
	[%]	-	-	-	-	-	-
<i>PyPSA</i>	[GWh]	+ 7	+ 7	+ 244	- 203	- 16	+ 178
	[%]	+ 10	+ 59	+ 51	- 2	- 7	+ 302

PyPSA kann mittels der Spitzenlasterzeuger Offshore-WKA (+ 51 %), GuD-Kraftwerk (+ 59 %) und Biogas-BHKW (+ 10 %) mehr Strom erzeugen und in den Batterien zwischenspeichern. Gleichzeitig muss weniger Energie durch Onshore-WKA (- 2 %) und PV (- 7 %) bereitgestellt werden. Die Folge ist eine Verringerung der notwendigen installierten Leistungen der RES (vgl. Tabelle 4-13), wodurch weniger Investitionskosten entstehen. Der Grund liegt in der Vernachlässigung der spezifischen Emissionen für den Batteriespeicher in *PyPSA*, weshalb dessen Verwendung hinsichtlich der übergeordneten Restriktion vorgezogen wird. Zu Zeiten geringen Bedarfs kann also mittels der Offshore-WKA mehr Strom erzeugt und für spätere Zeitpunkte mit niedrigen spezifischen Emissionen nutzbar gemacht werden.

Wird das Erzeugungsprofil der Komponenten in MW über die Zeitspanne eines Wintermonats (15. Januar – 15. Februar) aufgetragen, wird die Grundversorgung durch Onshore-WKA bestätigt.

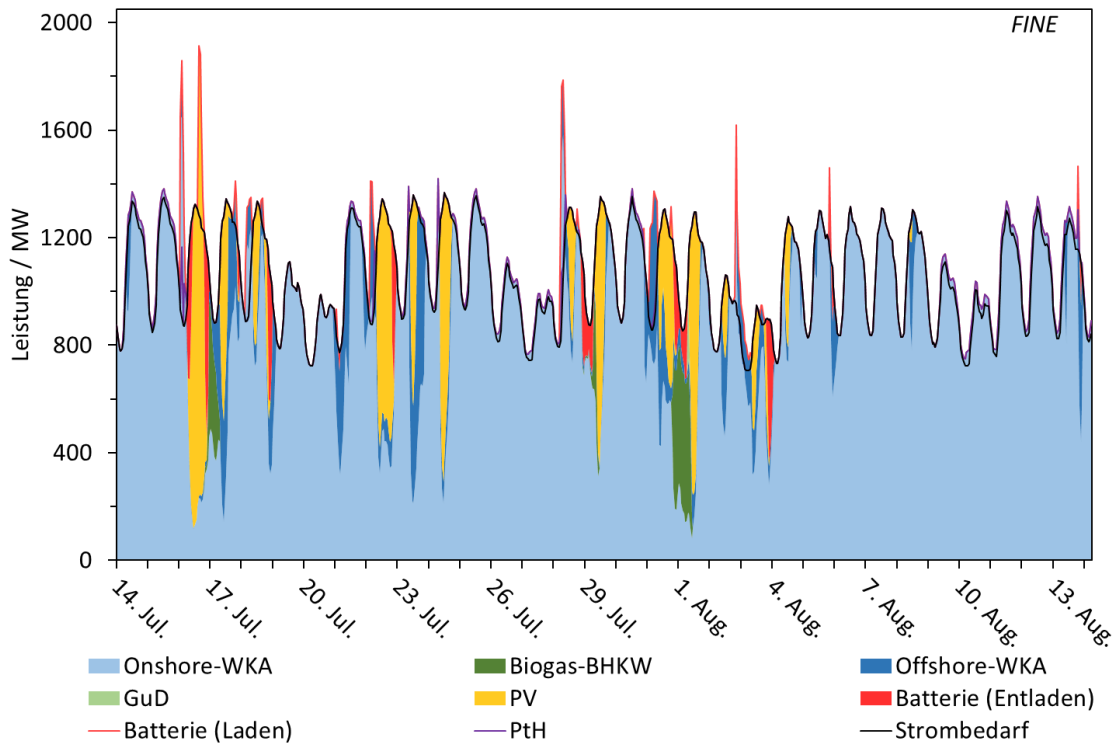


Abbildung 4-12: Erzeugungsprofil des Stromsektors eines Sommermonats für das Basisszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *FINE*.

Zu erkennen ist, dass der PV-Anteil wie erwartet steigt, bei gleichzeitig geringerer Bereitstellung von Strom durch das GuD-Kraftwerk. Da generell im Sommer weniger Wärme benötigt wird, fällt auch die Erzeugung durch die PtH-Anlage auf ein Minimum. Die sehr ähnlichen Profile von *Oemof* und *PyPSA* für je eine Sommer- und eine Winterperiode können Anhang 5 und Anhang 6 entnommen werden. *PyPSA* zeigt analog zu Tabelle 4-14, die Steigerung der Anteile durch Batterie und Offshore-WKA, bei gleichzeitiger Verringerung der Onshore-WKA. *Oemof* und *FINE* hingegen bedienen den Bedarf der Perioden deckungsgleich.

In Abbildung 4-10 ist außerdem zu erkennen, dass die gesamte erzeugte Strommenge im Basisszenario um 1 TWh (+ 10 %) höher ist, verglichen mit dem in Tabelle 3-18 festgelegten Bedarf. Die Erhöhung ist auf die Kopplung der Sektoren über die PtH-Anlage zurückzuführen, um den Wärmesektor zu versorgen. In Abbildung 4-13 mit den Wärmemengen für das Basisszenario aus Anhang 13 kann verdeutlicht werden, dass *FINE*, *Oemof* und *PyPSA* für den Wärmesektor Gemeinsamkeiten aufweisen.

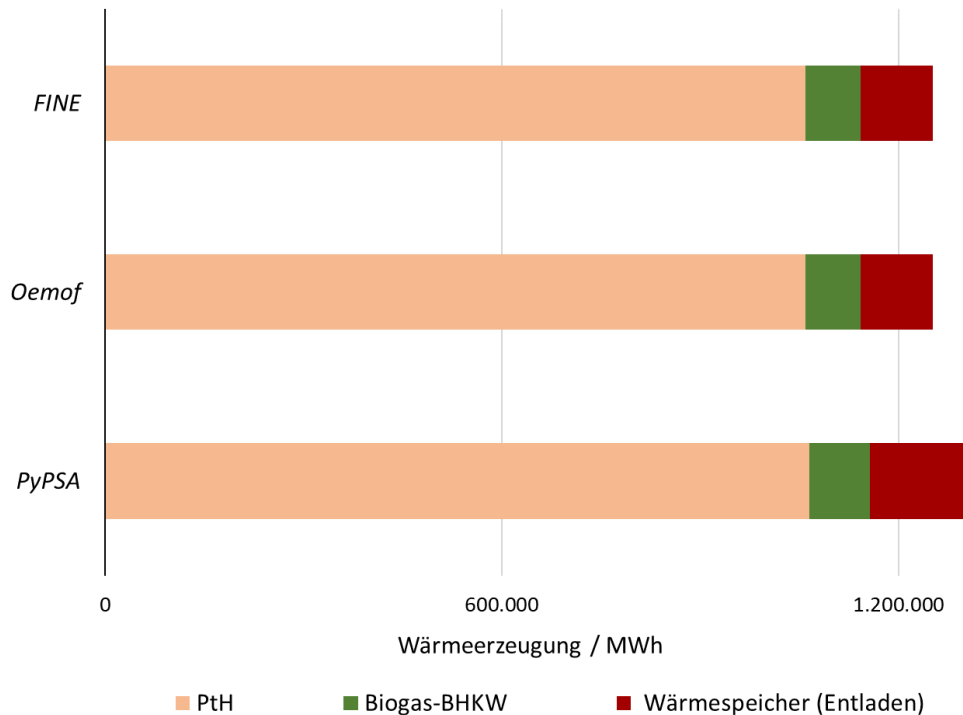


Abbildung 4-13: Zusammensetzung des Wärmesektors für das Basisszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *FINE*, *Oemof* und *PyPSA*.

PyPSA weist als einzigen Unterschied zwischen den Softwaremodellen die gesteigerte Wärmemenge, bereitgestellt durch den Wärmespeicher auf. Forciert wird dagegen in jeder Software die Kopplung der Sektoren, sodass über die PtH-Anlage 95 % der erforderlichen Wärme bereitgestellt werden kann. Detaillierter ist in Tabelle 4-15 zu erkennen, dass *FINE* und *Oemof* den Wärmebedarf identisch bedienen und *PyPSA* mehr Wärme bereitstellt.

Tabelle 4-15: Abweichung der bereitgestellten Wärmemengen von *Oemof* und *PyPSA* relativ zu *FINE* für das Basisszenario des *component_es* der *Tessif*-Bibliothek.

		Biogas-BHKW	PtH	Wärmespeicher (Entladen)
<i>Oemof</i>	[GWh]	-	-	-
	[%]	-	-	-
<i>PyPSA</i>	[GWh]	+ 9	+ 5	+ 39
	[%]	+ 10	< + 1	+ 35

Die drei verwendeten Komponenten Biogas-BHKW (+ 10 %), PtH-Anlage (< 1 %) und Wärmespeicher (+ 35 %) stellen in *PyPSA* mehr Wärme bereit, als in *Oemof* und *FINE*. Analog zu den Differenzen der Strombereitstellung (Tabelle 4-14), sind die Abweichungen im Wärmesektor auf die vernachlässigten spezifischen Emissionen von PtH-Anlage und Batteriespeicher zurückzuführen. Da sich die Ergebnisse von *PyPSA* und *Oemof* annähern, sobald die Emissionen vernachlässigt werden, ist auch zwischen *PyPSA* und *FINE* von diesem Verhalten auszugehen.

Das Erzeugungsprofil der Komponenten des Wärmesektors für einen Wintermonat (15. Januar – 15. Februar) bestätigt die Grundlastversorgung durch die PtH-Anlage und lässt Schlussfolgerungen zu, wie der Wärmespeicher dimensioniert wird.

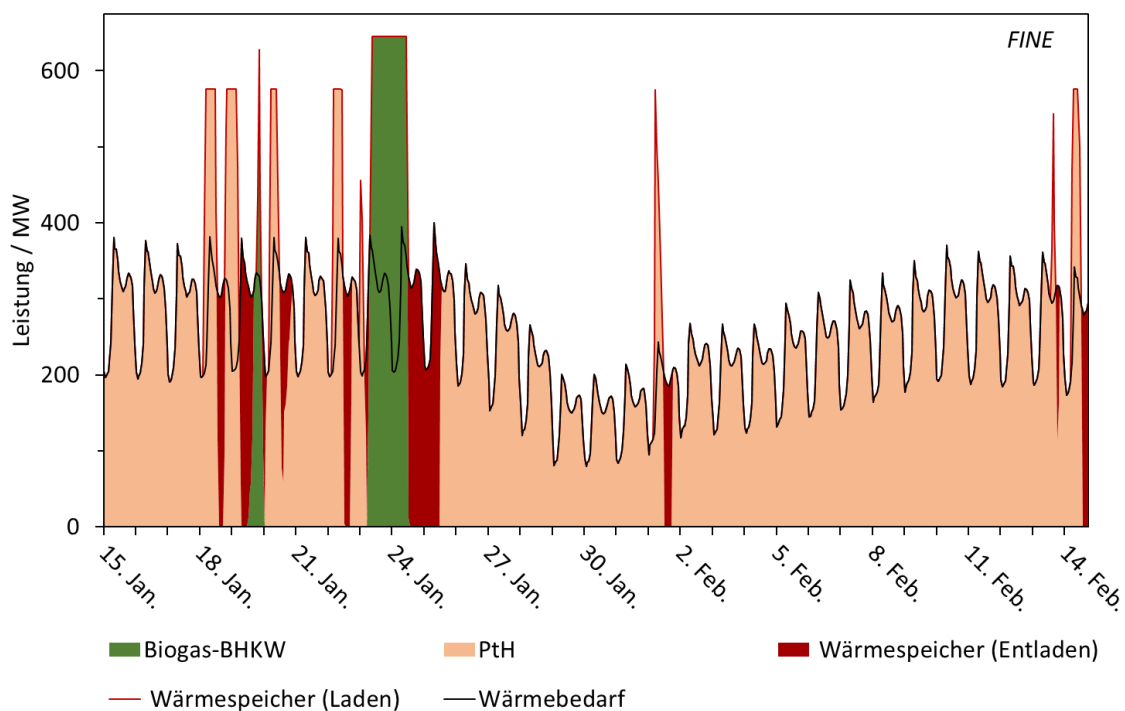


Abbildung 4-14: Erzeugungsprofil des Wärmesektors eines Wintermonats für das Basisszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *FINE*.

Wird Strom durch das Biogas-BHKW aufgrund der geringeren zur Verfügung stehenden Windleistung erzeugt, muss gleichzeitig auch die dadurch erzeugte Wärme verwendet werden (24. Januar). Diese wird genutzt, um den bestehenden Bedarf zu decken oder den Wärmespeicher zu laden. Es kann davon ausgegangen werden, dass der Wärmespeicher für diesen Zweck dimensioniert wird. In Zeitabschnitten, in denen die fluktuierenden RES

den Wärmebedarf nicht decken können, wird auf das Biogas-BHKW zurückgegriffen, da aufgrund der spezifischen Emissionen dieses die ökologischste Komponente ist (vgl. Tabelle 3-19). Dieses Vorgehen wird durch alle drei Softwares gleichermaßen durchgesetzt. Die mittels *Oemof* und *PyPSA* nahezu identischen Erzeugungsprofile für je einen Sommer- und einen Wintermonat können Anhang 7 und Anhang 8 entnommen werden.

Abschließend wird durch Anpassung des Basisszenarios eine Sensitivitätsanalyse durchgeführt. Wie eingangs beschrieben, gleichen sich die Ergebnisse von *Oemof* und *PyPSA* an, sobald die spezifischen Emissionen von Batterie und PtH-Anlage aus dem Energiesystem entfernt werden [41]. Dies soll auch für *FINE* geprüft werden, um die Parallelen aufzuzeigen und die Ergebnisse zu bestätigen. Die entstehenden Gesamtergebnisse der jeweiligen Optimierungen können der folgenden Tabelle 4-16 entnommen werden.

Tabelle 4-16: Gesamtergebnisse zu Kosten und Emissionen unter Vernachlässigung der spezifischen Emissionen von Batterie und PtH-Anlage für das Basisszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *FINE*, *Oemof* und *PyPSA*.

	<i>FINE</i>	<i>Oemof</i>	<i>PyPSA</i>
Emissionen [kt]	250	250	250
Gesamtkosten [10^3 €]	37.793.513	37.793.513	37.793.513
Betriebskosten [10^3 €]	820.206	820.206	820.206
Investitionskosten [10^3 €]	36.973.308	36.973.307	36.973.307

Die Gesamt- sowie Betriebs- und Investitionskosten von *FINE* gleichen sich denen von *Oemof* und *PyPSA* an und das Emissionslimit wird jeweils eingehalten. Daher kann die Aussage getroffen werden, dass die Softwaremodelle in Folge der Anpassungen das betrachtete Energiesystem identisch auslegen. Die Untersuchung des Basisszenarios ist damit beendet und analysierten Gemeinsamkeiten und Unterschiede der Softwaremodelle bestätigen die korrekte Integration von *FINE* in *Tessif*, für die enthaltenen Komponenten und Parameter des ausbaufähigen Energiesystems.

5 Zusammenfassung und Fazit

Abschließend werden die signifikanten Ergebnisse dieser Arbeit zusammengefasst und ein Fazit daraus abgeleitet. Der Fokus liegt dabei auf der korrekten Integration von *FINE* in *Tessif* sowie auf dem Vergleich der Softwares und den jeweilig erzielten Ergebnissen der Szenarioanalyse für ein Beispielenergiesystem.

5.1 Zusammenfassung

Das komponentenbasierte Energiesystem (vgl. Abbildung 3-6) der *Tessif*-Bibliothek wurde ausgewählt, um die Fähigkeiten von *FINE* in dem Kontext von *Tessif* zu demonstrieren. Durch die drei in *Tessif* integrierten Softwaremodelle wurde das Beispielenergiesystem optimiert, um anschließend signifikante Ergebnisse zu Kosten, Emissionen, Erzeugungs- und Lastprofilen sowie zu installierten Leistungen bzw. Kapazitäten zu extrahieren.

Zunächst wurde der Einfluss durch die Aggregation der Zeitreihen mittels *tsam* für die Integration von *FINE* in *Tessif* untersucht. Für das Referenzszenario konnte mittels der Aggregation eine Reduktion der Optimierungszeit um 3,05 min (- 99,88 %) realisiert werden, wobei gleichzeitig Gesamtkosten um 2,05 Mio. € (< - 1 %) unterschätzt wurden.

Die Zeitersparnis für das Basisszenario lag in Folge der Aggregation bei 14,93 min (- 98,68 %). Die berechneten Gesamtkosten wurden dabei für ein ausbaufähiges Energiesystem mit 7,61 Mrd. € (- 18 %) deutlicher als im Referenzszenario unterschätzt. Generell war für das untersuchte Energiesystem die Aggregation nicht erforderlich, da die Optimierungszeiten ohne aggregierter Datengrundlage in einem akzeptablen und mit *Oemof* und *PyPSA* vergleichbaren Bereich lagen (vgl. Tabelle 4-6).

Eine Szenarioanalyse mit dem ausgewählten Energiesystem wurde gleichermaßen mittels der drei in *Tessif* integrierten Softwares durchgeführt, um anhand der Optimierungsergebnisse die korrekte Integration von *FINE* zu prüfen. Für das Referenzszenario lässt sich zusammenfassen, dass die drei Softwaremodelle deutliche Gemeinsamkeiten aufweisen. Die jeweilig berechneten Gesamtkosten und Emissionen stimmen bis auf wenige Prozent überein (vgl. Tabelle 4-7) und Unterschiede konnten lediglich in der Zusammensetzung der Energiebereitstellung gefunden werden. Die Optimierung durch *FINE* berechnete im

Vergleich zu *Oemof* eine größere bereitgestellte Strommenge durch das Steinkohle-Kraftwerk und gleichzeitig eine verringerte Menge durch Photovoltaik (vgl. Tabelle 4-8). Dies konnte auf die identischen Betriebskosten der beiden Komponenten (vgl. Tabelle 3-20) zurückgeführt werden, sodass sie in ihrer wirtschaftlichen Effizienz gleichwertig sind. Auch im Wärmesektor des Referenzszenarios entstanden Differenzen in der Energiebereitstellung zwischen den Softwaremodellen (vgl. Tabelle 4-9). Die Analyse ergab, dass diese Abweichungen in Folge der erhöhten Verluste durch Selbstentladung des Wärmespeichers resultieren. Während *Oemof* und *PyPSA* identische Speicherverluste von 29,63 % aufwiesen, stiegen diese in *FINE* auf 30,21 %. Der Grund dafür konnte in den angepassten Speicherfüllständen zu Beginn und am Ende der untersuchten Zeitspanne gefunden werden. Die Integration von *FINE* sieht es nicht vor, dass der initiale bzw. finale Speicherstand festgelegt werden kann. Deshalb wurde während der Optimierung durch die Software ein passender Speicherstand gewählt, um den zeitnahen Bedarf möglichst effektiv zu decken. Sofern über *Tessif* auch für *Oemof* und *PyPSA* der initiale bzw. finale Speicherstand variabel definiert wird, ist jedoch davon auszugehen, dass identisches Verhalten eintritt.

FINE und *Oemof* berechneten für das Basisszenario identische Gesamtkosten (vgl. Tabelle 4-12), während *PyPSA* eine Differenz von 4,5 Mrd. € (– 11 %) aufwies. Darüber hinaus verfehlte *PyPSA* das Emissionslimit um 15 kt (+ 6 %). Weitere Gemeinsamkeiten für *FINE* und *Oemof* zeigten sich in den Energiebereitstellungen durch alle Komponenten, die identisch ausfielen, während *PyPSA* erneut Unterschiede aufzeigte. Die bisherige Forschung mittels *Tessif* zu dem komponentenbasierten Energiesystem konnte zeigen, dass *PyPSA* die Optimierung unter Vernachlässigung der spezifischen Emissionen von Batterie und PtH-Anlage durchführt [41]. Während einer abschließenden Sensitivitätsanalyse wurden die spezifischen Emissionen dieser Komponenten entfernt, um die identische Ausgangsbasis für alle drei Softwaremodelle zu schaffen. Ein erneuter Vergleich bestätigte, dass in Folge dieser Vernachlässigung identische Gesamtergebnisse für *Oemof*, *PyPSA* und *FINE* entstehen (vgl. Tabelle 4-16).

5.2 Fazit

Ziel dieser Arbeit war es, eine weitere Software zur Modellierung und Optimierung von Energiesystemen in eine Schnittstelle für einen systematischen Vergleich zu integrieren. Das Framework *Transforming Energy Supply System (Modelling) Frameworks (Tessif)* ermöglichte bisher die Verwendung von den Softwaremodellen *Oemof* und *PyPSA*. Das *Framework for Integrated Energy System Assessment (FINE)* wurde identifiziert und durch die individuelle Entwicklung von zwei Funktionen in *Tessif* integriert. Die erste Funktion dient zur initialen Übersetzung von Energiesystemen (*es2es*, vgl. Kapitel 3.2.1) und die zweite für eine standardisierte Darstellung der Ergebnisse aus der jeweiligen Optimierung (*es2mapping*, vgl. Kapitel 3.2.2). Insbesondere die speziell für *FINE* entwickelte, automatisierte Transformation von indirekten Komponenten stellte eine Herausforderung dar. Die Integration von *FINE* ermöglicht es nun Quellen und Speichern spezifische Emissionen zuzuordnen und es kann ein Konnektor als Verbindung von Potentialebenen definiert werden. Die Fähigkeiten von *FINE* und *Tessif* erlauben es so zukünftige Analysen komplexer, realitätsnaher Energiesysteme durchzuführen. Anhand eines speziell für die Prüfung der Integration von *FINE* ausgewählten Energiesystems konnte gezeigt werden, dass nahezu identische Ergebnisse von *FINE*, *Oemof* und *PyPSA* berechnet werden. Entstandene Unterschiede wurden identifiziert und analysiert, sodass der Ursprung der Abweichungen aufgeklärt werden konnte. Das System umfasst zudem ein breites Spektrum an Komponenten und Parametern, weshalb insgesamt die erfolgreiche Integration von *FINE* in *Tessif* demonstriert wurde.

Die Fortschritte dieser Arbeit erweitern die Fähigkeiten und den Nutzen von *Tessif*, um einen Beitrag für die computergestützte Modellierung und Optimierung von Energiesystemen zu leisten.

6 Ausblick

Das in dieser Arbeit untersuchte Energiesystem für den systematischen Vergleich der drei in *Tessif* integrierten Softwaremodelle deckt bereits ein großes Spektrum an möglichen Komponenten und zugehörigen Parametern ab. Aufgrund der Architektur des komponentenbasierten Energiesystems mit vereinfachter Betrachtung der Transportinfrastruktur ist jedoch die Notwendigkeit eines Konnektors entfallen. Dieser würde zum Beispiel Stromnetze unterschiedlicher Potentialebenen verbinden, sodass ein realitätsnäheres System entwickelt und optimiert werden könnte. Zukünftige Forschung mittels *Tessif* sollte dies berücksichtigen und die Ergebnisse von Konnektoren kritisch hinterfragen. Empfohlen wird daher ein weiterer Vergleich der Softwaremodelle mit Hilfe des netzbasierten Energiesystems (*Grid_es*), enthalten in der *Tessif*-Bibliothek, wo die Verbindung von unterschiedlichen Potentialebenen explizit beschrieben wird. Zukünftig noch zu prüfende technische Parameter der weiteren Komponenten sind die Begrenzungen der Lastwechselrate (*flow_gradients*) sowie der insgesamt zur Verfügung gestellten Energiemenge (*accumulated_amounts*). Ein weiteres in der *Tessif*-Bibliothek vorhandenes Beispiel ist das Hamburger Energiesystem, das die noch zu untersuchenden Parameter enthält. So kann mit Hilfe der bereits vorhandenen Energiesysteme der *Tessif*-Bibliothek ein möglichst umfassender Softwarevergleich durchgeführt werden. Erste Durchläufe der Optimierungen für die Beispiele des *Grid_es* und des Hamburger Energiesystems zeigen, dass die Gesamtergebnisse von *FINE*, *Oemof* und *PyPSA* in der gleichen Größenordnung liegen und daher vergleichbar sind.

Um die Fähigkeiten von *Tessif* zu erweitern, können verschiedene Maßnahmen sinnvoll sein. Das neu integrierte Softwaremodell *FINE* sollte erneut auf technische und ökonomische Parameter untersucht werden, die *Tessif* und den weiteren Softwaremodellen hinzugefügt werden können. Die Investitionen für den Ausbau einer Komponente entstehen beispielsweise direkt und werden ausschließlich für die untersuchte Zeitspanne angerechnet. *FINE* kann die ökonomische Lebensdauer und einen Zinssatz mit einbeziehen, sodass die Kostenberechnung dann realitätsnäher wäre. Eine weitere Maßnahme wäre die Modellierung von Energieumwandlungen mittels Nichtlinearitäten. Der effektive Betrieb

von Kraft-Wärme-Kopplung zeichnet sich durch ein nicht fest vorgegebenes Erzeugungsprofil mit variabler Wärmeauskopplung aus. In *FINE* ist für Teillastverhalten von Energieumwandlungen bereits eine spezielle Komponente vorgesehen, die zukünftig in *Tessif* integriert werden könnte.

Damit generell weitere Gemeinsamkeiten und Unterschiede der Softwaremodelle herausgearbeitet werden können, sollten weitere Energiesysteme entwickelt werden. Interessant wäre insbesondere eine Langzeitstudie, die eine Zeitspanne von mehreren Jahren betrachtet. Für diesen Fall könnte die in *FINE* und *Tessif* zusätzlich vorhandene Möglichkeit für die Aggregation der Zeitschritte von Nutzen sein und die erforderliche Berechnungszeit verringern. Ein weiteres zu entwickelndes Energiesystem sollte den Ansatz der „Grünen Wiese“ prüfen. Dieser geht von einem natürlichen Ausgangszustand aus, weshalb speziell die zu tätigen Investitionskosten vergleichbar wären. Neben neu entwickelten Energiesystemen sollten weitere Softwaremodelle integriert werden, um weitere Gemeinsamkeiten und Unterschiede zu verdeutlichen.

Schließlich gilt es mit den Erkenntnissen bezüglich der integrierten Softwaremodelle eine Empfehlung auszusprechen, in welchem Forschungsfeld der größtmögliche Nutzen erzielt werden kann. Vor allem der Ansatz der freien Zugänglichkeit sowie die intuitive, interdisziplinäre Bedienung zeichnen *Tessif* aus. Dadurch soll zukünftig Entwicklern und Anwendern aus allen energietechnischen Bereichen ein schneller Zugang zu Modellierungs- und Optimierungssoftware von Energiesystemen ermöglicht werden. Langfristig wird damit ein Beitrag geleistet, die unterschiedlichen Transformationspfade des Energiesystems effizient zu untersuchen und einen multiplen Nutzen zu generieren.

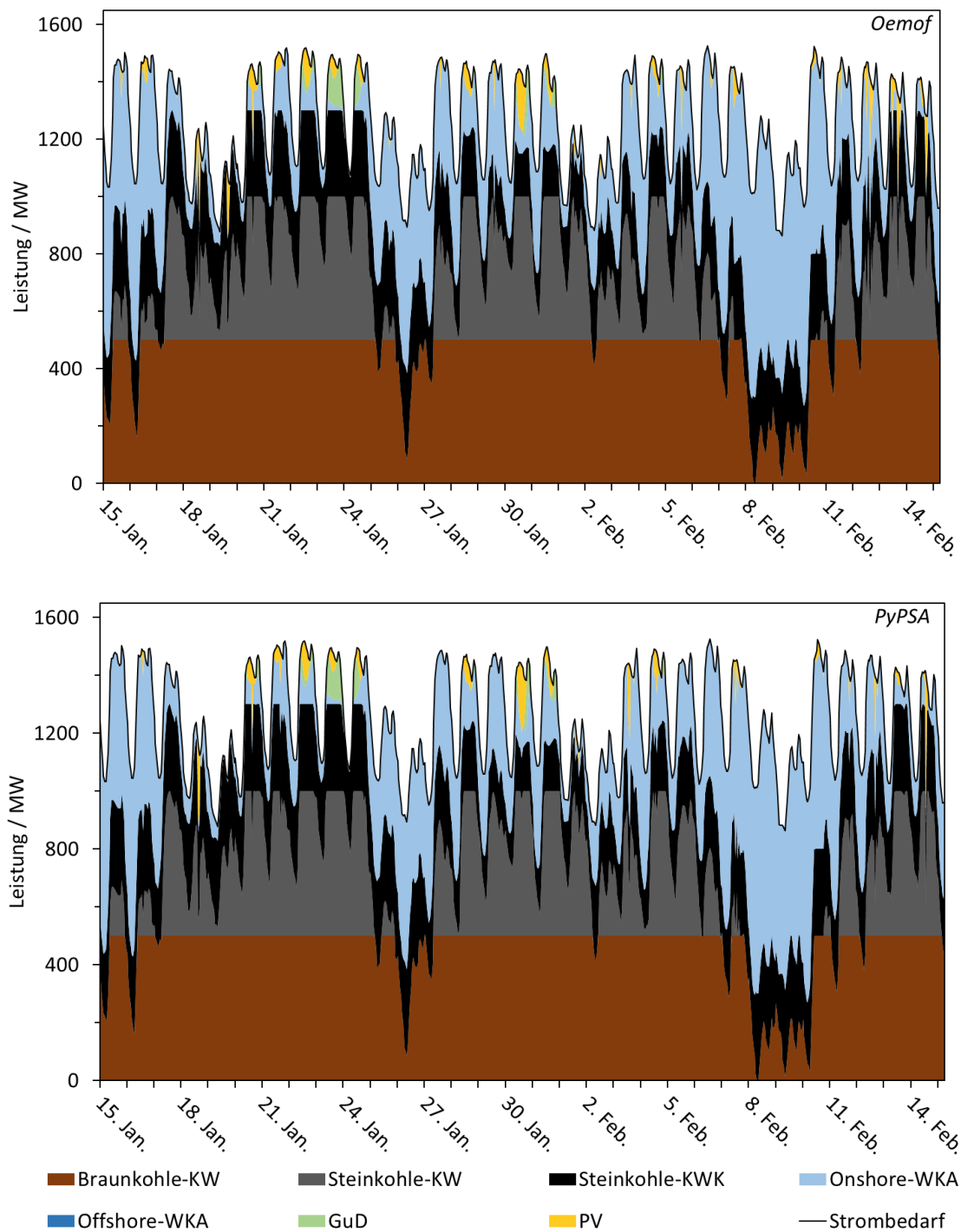
Literaturverzeichnis

- [1] Die Bundesregierung - Bundeskanzleramt: *Datenstrategie der Bundesregierung* URL: www.bundesregierung.de/publikationen (Aufgerufen am: 27.01.21).
- [2] Liggesmeyer, P., Rombach, D., Bomarius, F.: *Smart Energy*, in *Digitalisierung: Schlüsseltechnologien für Wirtschaft und Gesellschaft*, Fraunhofer-Forschungsfokus (Hrsg: R. Neugebauer), Springer Vieweg. Berlin, Heidelberg, **2018**.
- [3] Chang, M., et al.: *Trends in tools and approaches for modelling the energy transition*, In: *Applied Energy* 290, S. 116731, **2021**.
- [4] Ringkjøb, H.-K., Haugan, P., Solbrekke, I.: *A review of modelling tools for energy and electricity systems with large shares of variable renewables*, In: *Renewable and Sustainable Energy Reviews* 96, S. 440 – 459, **2018**.
- [5] Gils, H.-C., et al.: *Modeling flexibility in energy systems – a scenario-based comparison of power sector models*, In: *Renewable and Sustainable Energy Reviews* 26.03.21.
- [6] Connolly, D., et al.: *A review of computer tools for analysing the integration of renewable energy into various energy systems*, In: *Applied Energy* 87, S. 1059 – 1082, **2010**.
- [7] Kastens U., Kleine Büning H., *Modellierung: Grundlagen und formale Methoden*, 4 Aufl., Hanser eLibrary, Carl Hanser Verlag, München, **2018**.
- [8] *Softwaresysteme und ihre Modellierung: Grundlagen, Methoden und Techniken*, SpringerLink Bücher, Springer Berlin Heidelberg, Berlin, Heidelberg, **2006**.
- [9] Falke, T., et al.: *Multi-objective optimization and simulation model for the design of distributed energy systems*, In: *Applied Energy* 184, S. 1508 – 1516, **2016**.
- [10] Farzaneh H., *Energy Systems Modeling*, Springer Singapore, Singapore, **2019**.
- [11] Pfenninger, S., et al.: *Opening the black box of energy modelling: Strategies and lessons learned*, In: *Energy Strategy Reviews* 19, S. 63 – 71, **2018**.
- [12] Hourcade, J.: *Estimating the Cost of Mitigating Greenhouse Gases.*, In: *Climate Change 1995: Economic and Social Dimensions of Climate Change 1996*, S. 263 – 296, **1995**.
- [13] Pfluger B., Tersteegen B., Franke B.: *Langfristszenarien für die Transformation des Energiesystems in Deutschland: Modul 3: Referenzszenario und Basisszenario* Studie im Auftrag des Bundesministeriums für Wirtschaft und Energie, **2017**.
- [14] Heffels, T., *Kraftwerks- und Speicherbedarf bei hohen Anteilen erneuerbarer Energien*, Dissertation. Fakultät für Wirtschaftswissenschaften, Karlsruher Instituts für Technologie (KIT), **2015**.
- [15] COIN-OR Foundation, *COIN-OR Branch-and-Cut solver: Cbc: Version 2.10.5*.
- [16] Free Software Foundation: *GLPK (GNU Linear Programming Kit)* URL: <https://github.com/xypron/glpk-cli> (Aufgerufen am: 09.11.21).

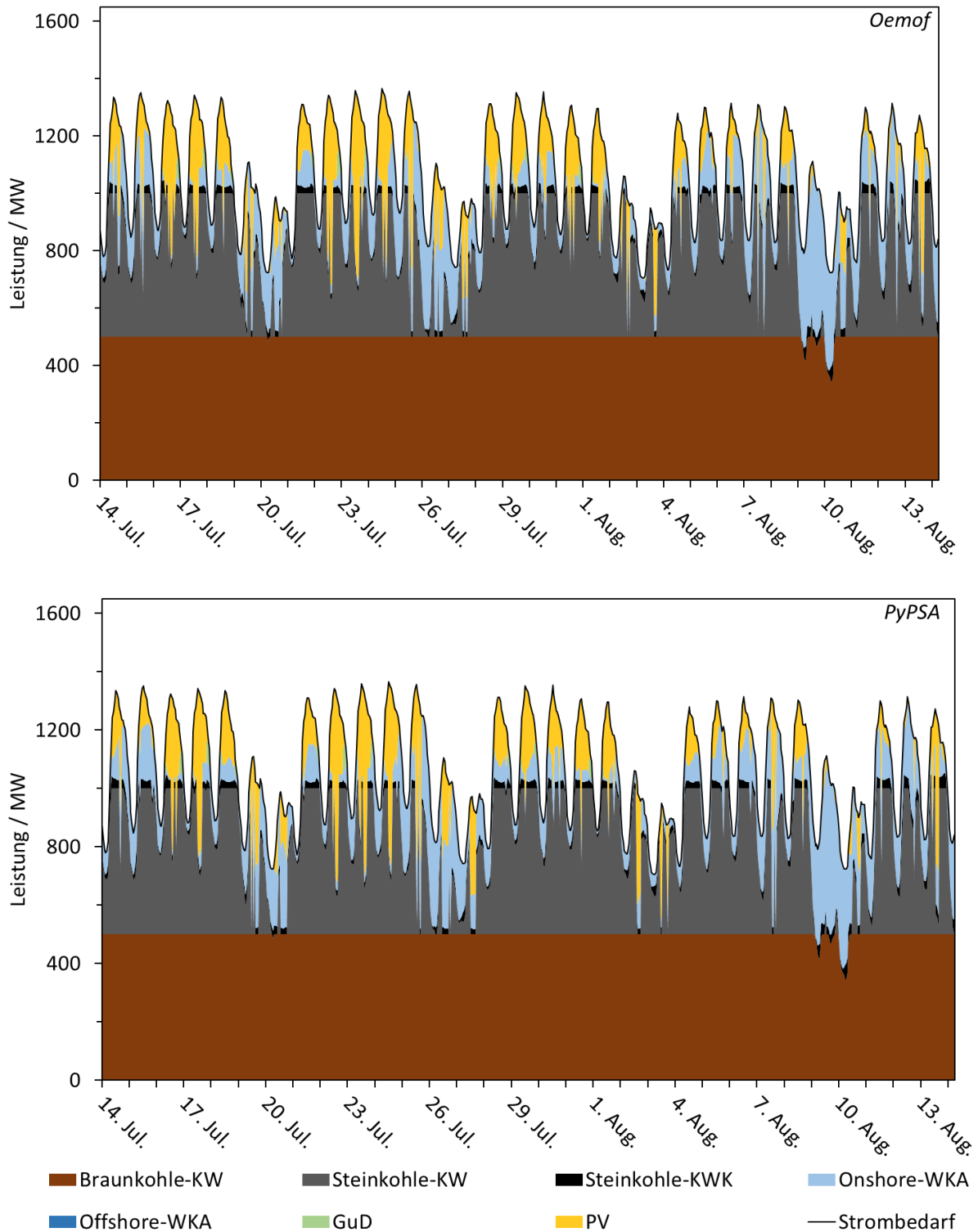
- [17] Bauer, D., et al.: *Modellbasierte Optimierung von Energiesystemen*, In: Cunningham, D. W., Hofstedt, P., Meer, K. & Schmitt, I. (Hrsg.), INFORMATIK 2015. Bonn: Gesellschaft für Informatik e.V, S. 137 – 149, **2015**.
- [18] Kallrath J., *Lineare Optimierung und Netzwerkoptimierung: Zweisprachige Ausgabe Deutsch Englisch 2., verbesserte Auflage*, Springer eBook Collection, Vieweg, Wiesbaden, **2006**.
- [19] Unger T., *Lineare Optimierung: Modell, Lösung, Anwendung*, Springer eBook Collection, Vieweg+Teubner, Wiesbaden, **2010**.
- [20] Liebl F., *Simulation: Problemorientierte Einführung*, 2 Aufl., Oldenbourg Wissenschaftsverlag, Berlin, Boston, **1995**.
- [21] Kallrath J., *Gemischt-ganzzahlige Optimierung: Modellierung in der Praxis: Mit Fallstudien aus Chemie, Energiewirtschaft, Papierindustrie, Metallgewerbe, Produktion und Logistik*, 2 Aufl., Springer eBook Collection, Springer Vieweg, Wiesbaden, **2013**.
- [22] Ommen, T., Markussen, W., Elmegaard, B.: *Comparison of linear, mixed integer and non-linear programming methods in energy system dispatch modelling*, In: Energy 74, S. 109 – 118, **2014**.
- [23] Wilkerson, J., et al.: *Comparison of integrated assessment models: Carbon price impacts on U.S. energy*, In: Energy Policy 76, S. 18 – 31, **2015**.
- [24] Priesmann, J., Nolting, L., Praktijnjo, A.: *Are complex energy system models more accurate? An intra-model comparison of power system optimization models*, In: Applied Energy 255, S. 113783, **2019**.
- [25] Misconel, S., et al.: *Systematic comparison of high-resolution electricity system modeling approaches focusing on investment, dispatch and generation adequacy*, In: Renewable and Sustainable Energy Reviews 153, S. 111785, **2021**.
- [26] Steinbrink, C., et al.: *Simulation-Based Validation of Smart Grids – Status Quo and Future Research Trends*, in *Industrial Applications of Holonic and Multi-Agent Systems*, Vol. 10444, Lecture Notes in Computer Science (Hrsg: V. Mařík, W. Wahlster, T. Stras-ser, P. Kadera), Springer International Publishing. Cham, **2017**.
- [27] *Openmod: Open Energy Modelling Initiative* URL: <https://openmod-initiative.org/> (Aufgerufen am: 27.01.22).
- [28] Mosaik Developer Group: *Mosaik: A flexible Smart Grid co-simulation framework* URL: <http://mosaik.offis.de/> (Aufgerufen am: 19.11.21).
- [29] Rohjans, S., et al.: *Gekoppelte Simulation komplexer Energiesysteme mittels MOSAIK und FMI*, In: at - Automatisierungstechnik 62, S. 325 – 336, **2014**.
- [30] Ammon M., *Tessif: Transforming Energy Supply System (Modelling) Frameworks*. Dokumentation URL: (unveröffentlicht), Institut für Energietechnik, Technische Universität Hamburg.

- [31] Oemof Developer Group: *A modular open source framework to model energy supply systems* URL: <https://oemof.org/> (Aufgerufen am: 27.01.22).
- [32] Brown, T., Hörsch, J., Schlachtberger, D.: *PyPSA: Python for Power System Analysis*, In: *Journal of Open Research Software* 6, **2018**.
- [33] FINE Developer Team, *FINE: A Framework for integrated energy system assessment* URL: <https://vsa-fine.readthedocs.io/en/latest/index.html> (Aufgerufen am: 28.11.21), Forschungszentrum Jülich.
- [34] Helmholtz-Gemeinschaft Deutscher Forschungszentren: *Energie System 2050* URL: <https://www.helmholtz.de/forschung/forschungsbereiche/energie/energie-system-2050/> (Aufgerufen am: 16.01.22).
- [35] Helmholtz-Gemeinschaft Deutscher Forschungszentren: *Helmholtz Energy Computing Initiative (HECI): Energie System 2050 (ES 2050)* URL: <https://www.helmholtz.de/forschung/forschungsbereiche/energie/energie-system-2050/heci/> (Aufgerufen am: 23.11.21).
- [36] Welder, L., et al.: *Spatio-temporal optimization of a future energy system for power-to-hydrogen applications in Germany*, In: *Energy* 158, S. 1130 – 1149, **2018**.
- [37] TSAM Developer Group, *tsam: Time Series Aggregation Module* URL: <https://tsam.readthedocs.io/en/latest/> (Aufgerufen am: 16.01.22), Forschungszentrum Jülich.
- [38] Kotzur, L., et al.: *Impact of different time series aggregation methods on optimal energy system design*, In: *Renewable Energy* 117, S. 474 – 487, **2018**.
- [39] Hoffmann, M., et al.: *A Review on Time Series Aggregation Methods for Energy System Models*, In: *Energies* 13, S. 641, **2020**.
- [40] Sandia National Laboratories: *Pyomo* URL: <http://www.pyomo.org/> (Aufgerufen am: 13.11.21).
- [41] Reimer, M., *Entwicklung eines Komponenten basierten Szenarios zum Vergleich von Free and Open Source Energiesystemmodellierungssoftware in Python*, Projektarbeit. Institut für Energietechnik, Technische Universität Hamburg, **2021**.

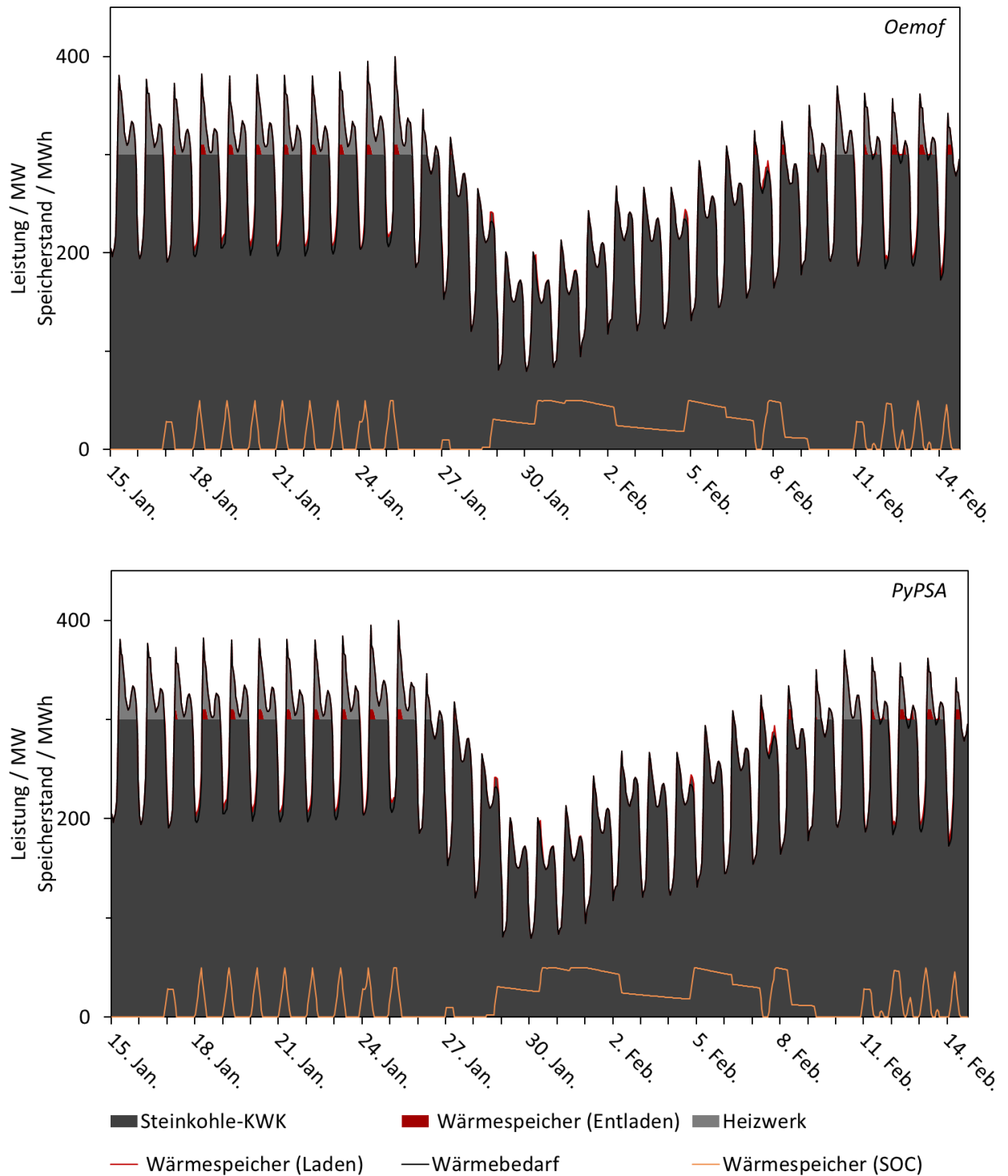
Anhang



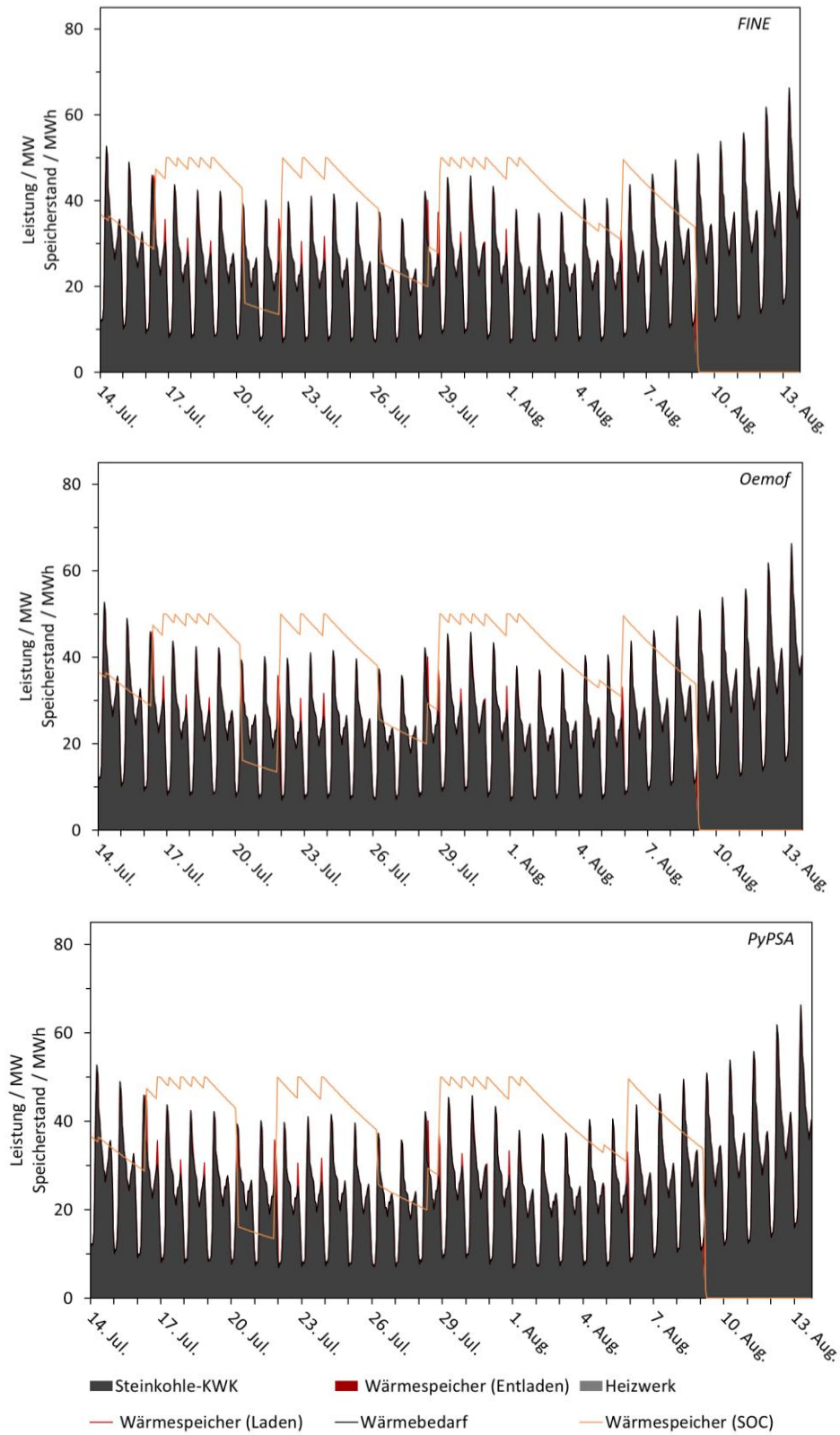
Anhang 1: Erzeugungprofile des Stromsektors eines Wintermonats für das Referenzszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *Oemof* und *PyPSA*.



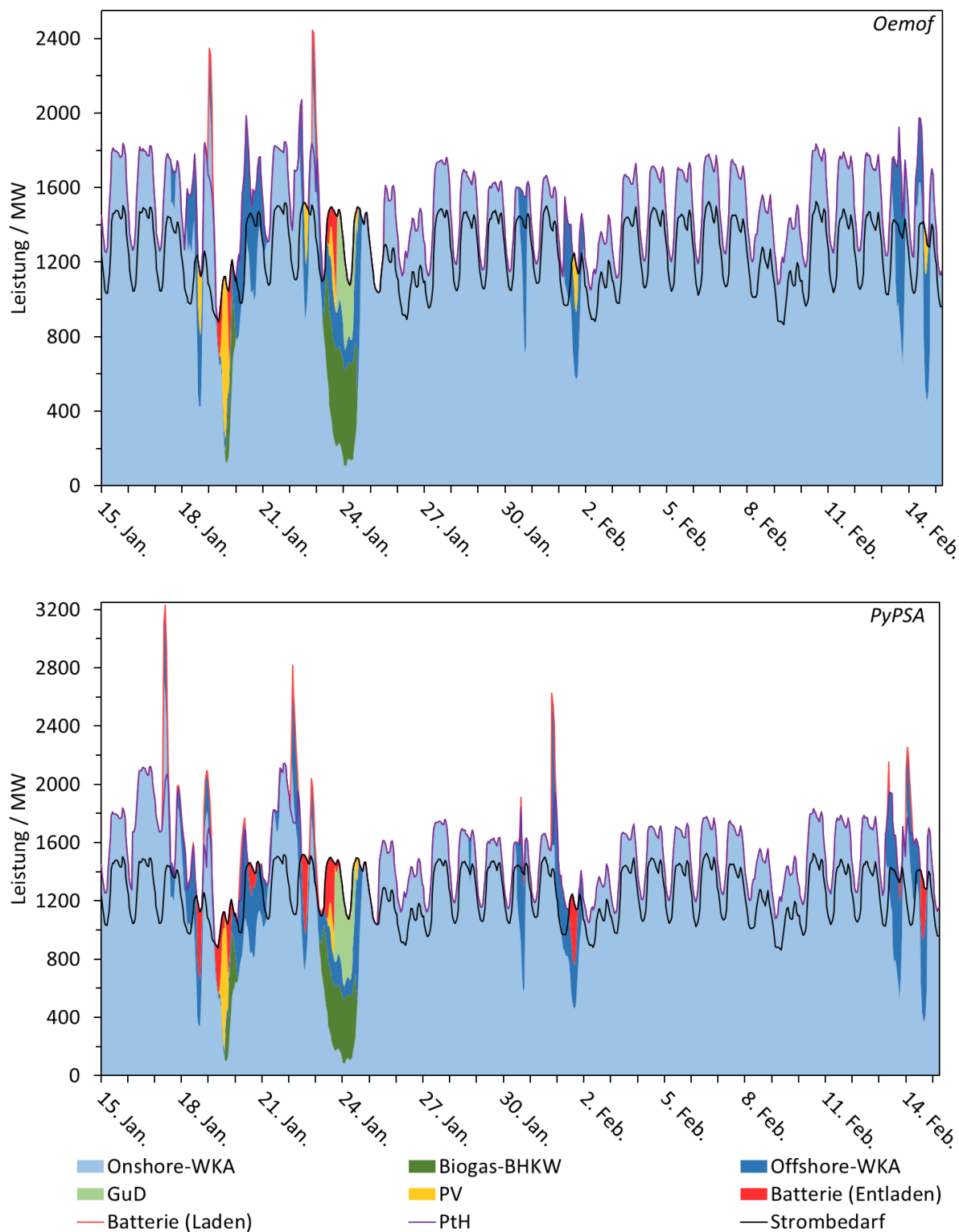
Anhang 2: Erzeugungprofile des Stromsektors eines Sommermonats für das Referenzszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *Oemof* und *PyPSA*.



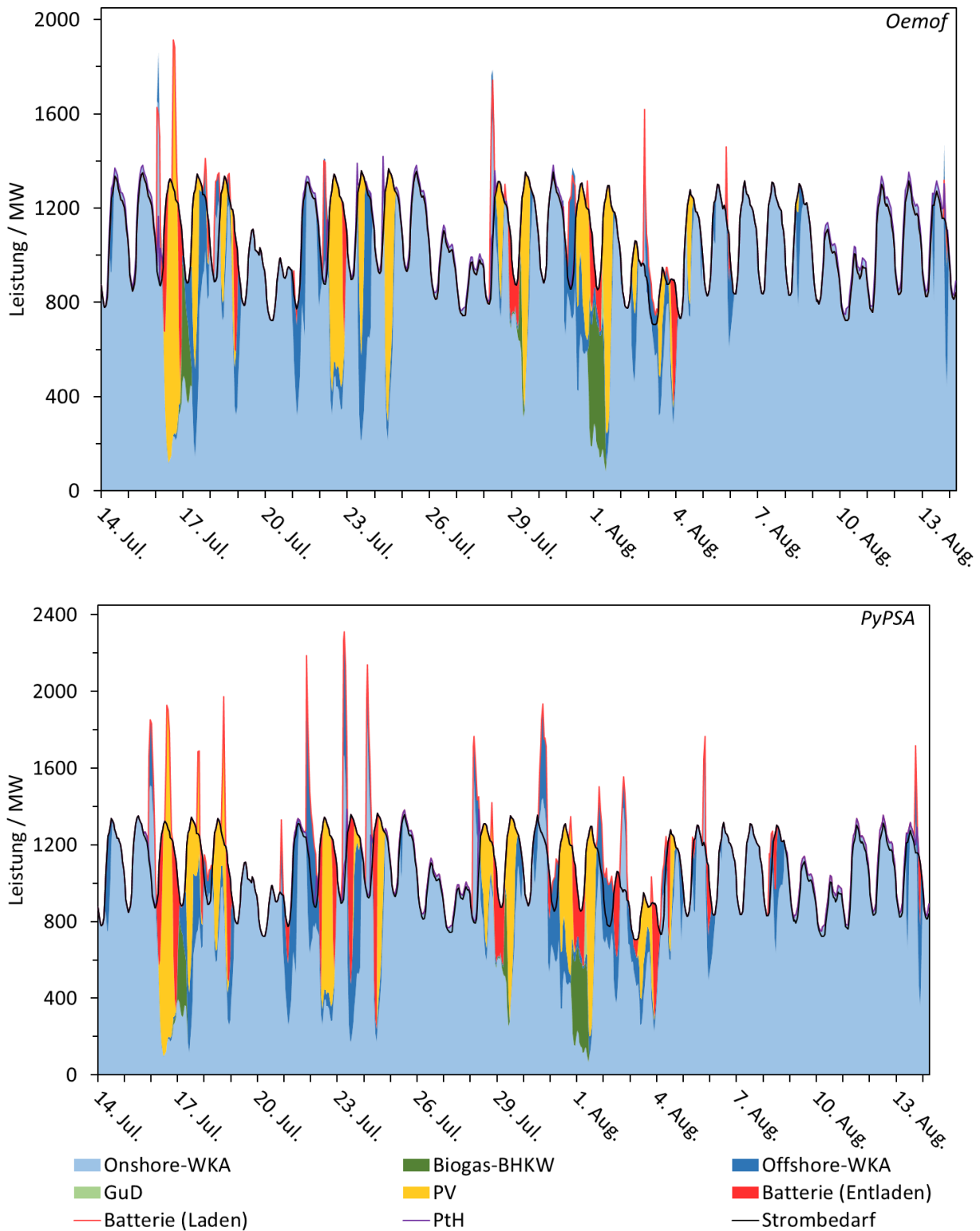
Anhang 3: Erzeugungprofile des Wärmesektors eines Wintermonats für das Referenzszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *Oemof* und *PyPSA*.



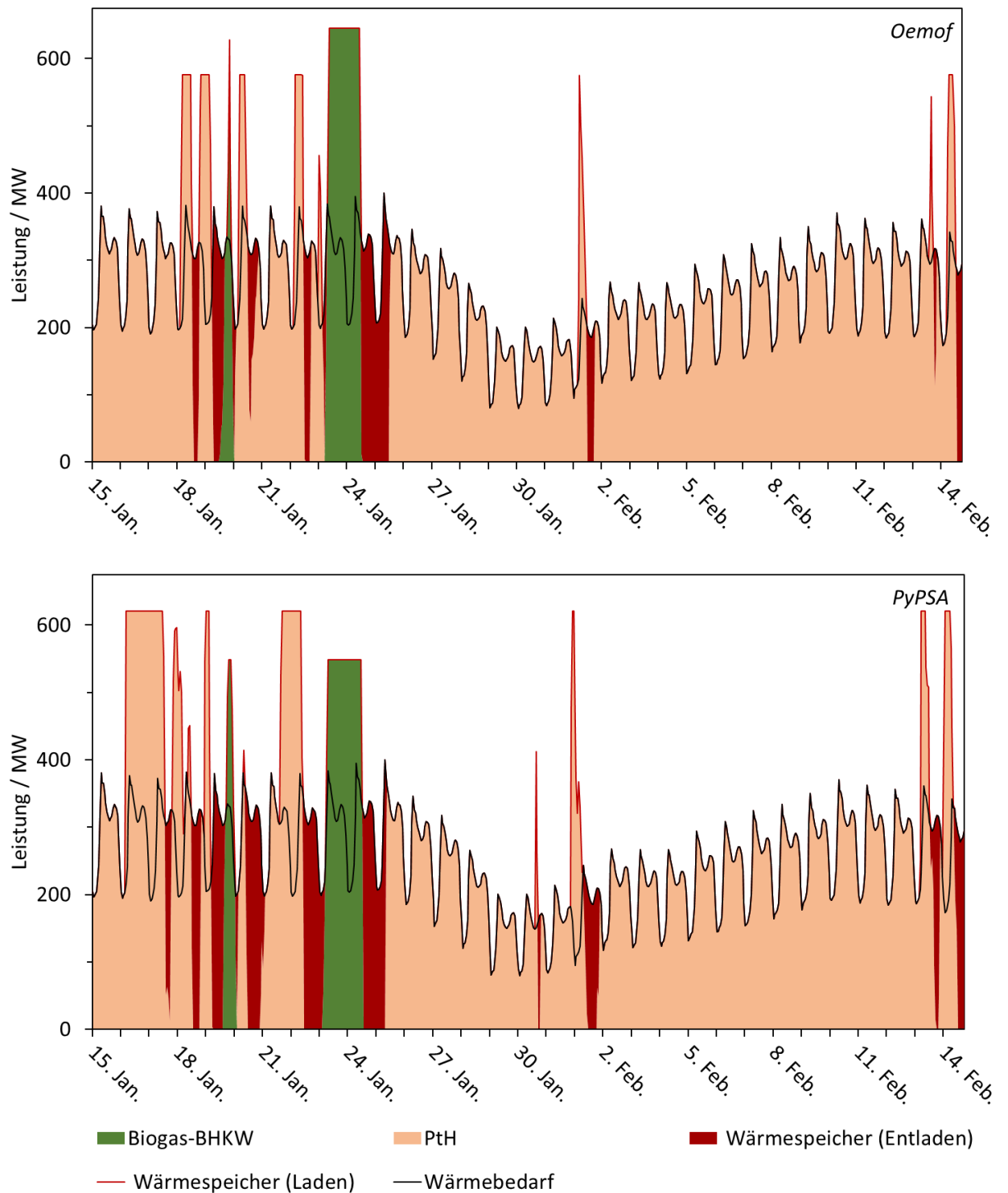
Anhang 4: Erzeugungprofile des Wärmesektors eines Sommermonats für das Referenzszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *Oemof* und *PyPSA*.



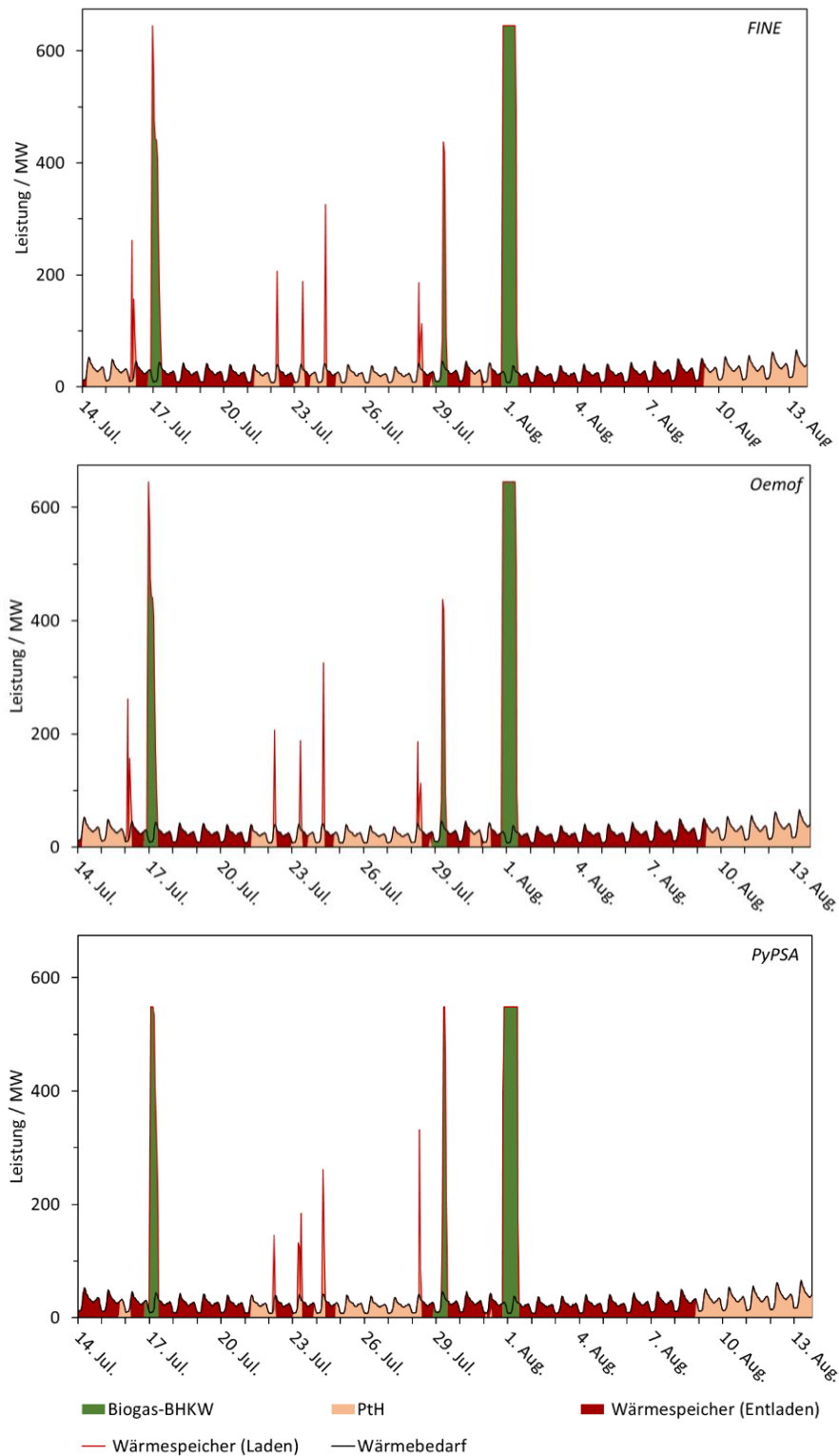
Anhang 5: Erzeugungprofile des Stromsektors eines Wintermonats für das Basisszenario des *componentes* der *Tessif*-Bibliothek optimiert durch *Oemof* und *PyPSA*.



Anhang 6: Erzeugungprofile des Stromsektors eines Sommermonats für das Basisszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *Oemof* und *PyPSA*.



Anhang 7: Erzeugungprofile des Wärmesektors eines Wintermonats für das Basisszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *Oemof* und *PyPSA*.



Anhang 8: Erzeugungprofile des Wärmesektors eines Sommermonats für das Basisszenario des *component_es* der *Tessif*-Bibliothek optimiert durch *Oemof* und *PyPSA*.

Anhang 9: Bereitgestellte Energiemengen jeder Komponente im Stromsektor optimiert durch FINE aggregiert mittels *tsam* für das *component_es* der *Tessif*-Bibliothek.

Szenario	Batterie (Entladen) [GWh]	Biogas-BHKW [GWh]	Offshore- WKA [GWh]	Onshore- WKA [GWh]	PV [GWh]
Referenz	-	-	-	2.081,97	386,21
Basis	194,53	73,39	38,39	10.779,57	23,18

Szenario	Steinkohle-KWK [GWh]	Steinkohle-KW [GWh]	Braunkohle-KW [GWh]	GuD [GWh]
Referenz	1.118,38	1.947,05	4.249,80	26,10
Basis	-	-	-	-

Anhang 10: Bereitgestellte Energiemengen jeder Komponente im Wärmesektor optimiert durch FINE aggregiert mittels *tsam* für das *component_es* der *Tessif*-Bibliothek.

Szenario	Biogas- BHKW [GWh]	Steinkohle-KWK [GWh]	Heiz- werk [GWh]	Wärmespeicher (Entladen) [GWh]	PtH-Anlage [GWh]
Referenz	-	1.118,38	2,29	1,41	-
Basis	91,74	-	-	96,38	1.058,91

Anhang 11: In Anspruch genommene Energiemengen jeder Komponente im Wärmesektor optimiert durch FINE aggregiert mittels *tsam* für das *component_es* der *Tessif*-Bibliothek.

Szenario	Batterie (Laden) [GWh]	Strombedarf [GWh]	PtH [GWh]	Wärme- bedarf [GWh]	Wärmespeicher (Laden) [GWh]
Referenz	-	9.809,51	-	1.116,16	5,92
Basis	229,94	9.809,51	1.069,61	1.116,16	130,87

Anhang 12: Bereitgestellte Energiemengen jeder Komponente im Stromsektor optimiert durch FINE, Oemof und PyPSA für das *component_es* der Tessif-Bibliothek.

Szenario	Software	Batterie (Entladen)	Biogas-BHKW	Offshore- WKA	Onshore- WKA	PV
		[GWh]	[GWh]	[GWh]	[GWh]	[GWh]
Referenz	<i>FINE</i>	-	-	-	2.081,34	386,31
	<i>Oemof</i>	-	-	-	2.081,33	388,67
	<i>PyPSA</i>	-	-	-	2.081,33	359,51
Basis	<i>FINE</i>	59,14	66,47	479,67	10.090,42	242,06
	<i>Oemof</i>	59,14	66,47	479,67	10.090,42	242,06
	<i>PyPSA</i>	237,49	73,41	723,18	9.886,99	226,10

Szenario	Software	Steinkohle-KWK	Steinkohle-KW	Braunkohle-KW	GuD
		[GWh]	[GWh]	[GWh]	[GWh]
Referenz	<i>FINE</i>	1.107,11	2.070,22	4.132,57	31,95
	<i>Oemof</i>	1.107,09	2.067,86	4.132,60	31,95
	<i>PyPSA</i>	1.107,09	2.097,02	4.132,60	31,95
Basis	<i>FINE</i>	-	-	-	11,51
	<i>Oemof</i>	-	-	-	11,51
	<i>PyPSA</i>	-	-	-	18,34

Anhang 13: Bereitgestellte Energiemengen jeder Komponente im Wärmesektor optimiert durch FINE, Oemof und PyPSA für das *component_es* der Tessif-Bibliothek.

Szenario	Software	Biogas-BHKW [GWh]	Steinkohle-KWK [GWh]	Heizwerk [GWh]	Wärmespeicher (Entladen) [GWh]	PtH-Anlage [GWh]
Referenz	<i>FINE</i>	-	1.107,11	10,10	2,43	-
	<i>Oemof</i>	-	1.107,09	10,10	2,45	-
	<i>PyPSA</i>	-	1.107,09	10,10	2,45	-
Basis	<i>FINE</i>	83,09	-	-	109,93	1.059,39
	<i>Oemof</i>	83,09	-	-	109,93	1.059,39
	<i>PyPSA</i>	91,76	-	-	148,89	1.064,80

Anhang 14: In Anspruch genommene Energiemengen durch jede Komponente im Stromsektor optimiert durch FINE, Oemof und PyPSA für das *component_es* der Tessif-Bibliothek.

Szenario	Software	Batterie (Laden) [GWh]	Strombedarf [GWh]	PtH [GWh]	Wärmebedarf [GWh]	Wärmespeicher (Laden) [GWh]
Referenz	<i>FINE</i>	-	9.809,51	-	1.116,16	3,48
	<i>Oemof</i>	-	9.809,51	-	1.116,16	3,48
	<i>PyPSA</i>	-	9.809,51	-	1.116,16	3,48
Basis	<i>FINE</i>	69,68	9.809,51	1.070,09	1.116,16	136,25
	<i>Oemof</i>	69,68	9.809,51	1.070,09	1.116,16	136,25
	<i>PyPSA</i>	280,44	9.809,51	1.075,56	1.116,16	189,29

Anhang 15: Benötigte Berechnungszeit als Mittelwerte über jeweils drei Optimierungen für das angepasste *component_es* der *Tessif*-Bibliothek für *Oemof*, *PyPSA* und *FINE*. (Vernachlässigung der Emissionen von Batterie und PTH-Anlage).

Software	Berechnungszeit / min	
	Referenz	Basis
<i>Oemof</i>	2,33	12,41
<i>PyPSA</i>	1,47	11,03
<i>FINE</i>	2,65	15,01