
Technische Hochschule Köln

Bachelorstudiengang Erneuerbare Energien

Bidirektionales Laden von Elektrofahrzeugen zur Verringerung der Netzbelastung von Industriestandorten

(überarbeitete und vollständig aktualisierte Fassung vom 6. April 2024)

Vorgelegt von: Jens Krätzschar
Adresse: Feldbergstr. 9
51105 Köln
jens.kraetzschmar@smail.th-koeln.de

eingereicht bei: Prof. Dr. Eberhard Waffenschmidt
Zweitgutachter: Prof. Dr. Ingo Stadler

Köln, 06.04.2024

Erklärungen

Name: Jens Krätzschar

Erklärung zum eigenständigen Verfassen

Ich versichere, die von mir vorgelegte Arbeit selbstständig verfasst zu haben. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Arbeiten anderer oder der Verfasserin/des Verfassers selbst entnommen sind, habe ich als entnommen kenntlich gemacht. Sämtliche Quellen und Hilfsmittel, die ich für die Arbeit benutzt habe, sind angegeben. Die Arbeit hat mit gleichem Inhalt bzw. in wesentlichen Teilen noch keiner anderen Prüfungsbehörde vorgelegen.

Köln, 06.04.2024

Ort, Datum

Rechtsverbindliche Unterschrift

Erklärung zur Veröffentlichung

Ich bin damit einverstanden, dass meine Abschlussarbeit ausgeliehen werden darf. Sie darf von meinem Betreuer im Internet veröffentlicht werden.

Köln, 06.04.2024

Ort, Datum

Rechtsverbindliche Unterschrift

Erklärung zu Bildrechten

Alle Bilder und Diagramme dieser Abschlussarbeit habe ich selbst erstellt.

Köln, 06.04.2024

Ort, Datum

Rechtsverbindliche Unterschrift

Danksagung

Ich möchte die Gelegenheit nutzen, um all jenen meinen aufrichtigen Dank auszusprechen, die mich bei der Erstellung dieser Arbeit unterstützt haben. Hierzu gehören insbesondere:

Michael König, der mich bei der Programmierung der Simulation maßgeblich unterstützt hat. Seine wertvollen Hinweise und klugen Ratschläge waren für die Fertigstellung dieser Arbeit von großer Bedeutung. Vielen Dank für die Zeit und Mühe, die du opfern musstest.

Ein besonderer Dank gebührt auch meiner Schwester Janett Krätzschar, die mir eine wertvolle Lektion vermittelt hat: Aufgeben ist keine Option. Sie hat mir gezeigt, dass man hartnäckig für seine Ziele kämpfen muss und dass Ausdauer der Schlüssel zum Erfolg ist.

Mein Dank gilt ebenso Prof. Eberhard Waffenschmidt, der während der gesamten Erarbeitungsphase dieser Abschlussarbeit als Betreuer fungierte und sich stets die Zeit nahm, meine Fragen eingehend zu diskutieren und zu beantworten.

Besonderer Dank gilt meinen Freunden und meiner Familie, die mich durch mein gesamtes Studium hindurch und insbesondere in der intensiven Phase der Anfertigung dieser Arbeit tatkräftig unterstützt haben. Ihre Erinnerungen, für ausreichend gesunde Nahrung zu sorgen, und die beharrliche Aufforderung, mir auch Zeiten der Ruhe zu gönnen, halfen mir dabei, die Balance zu wahren. Ihre unermüdliche Betreuung sorgte dafür, dass ich die nötigen Pausen einlegte, um neue Kraft zu schöpfen und meinen Geist zu erfrischen.

Kurzfassung/Abstract

Die vorliegende Bachelorarbeit widmet sich der Analyse und Bewertung der Einsatzmöglichkeiten von bidirektionalen Ladestationen für Elektrofahrzeuge an Industriestandorten.

Angesichts des steigenden Anteils erneuerbarer Energien im Stromnetz und der Zunahme von Elektrofahrzeugen untersucht diese Arbeit die Potenziale, die sich durch die Nutzung von Elektrofahrzeugbatterien als temporäre Energiespeicher ergeben. Im Fokus steht dabei die Fragestellung, inwieweit diese Technologie dazu beitragen kann, den Energieverbrauch in Hochlastzeiten zu optimieren und die Spitzenlast zu reduzieren.

Die Arbeit folgt einem klaren methodischen Ansatz, bei dem zunächst die technischen und theoretischen Grundlagen von Elektromobilität, Batterietechnologie und bidirektionalem Laden dargelegt werden. Darauf aufbauend wird ein Simulationsmodell entwickelt, welches verschiedene Einsatzszenarien an Industriestandorten abbildet. Die Modellierung berücksichtigt dabei sowohl technische als auch wirtschaftliche Parameter und zielt darauf ab, realistische Prognosen über den Einfluss von Elektrofahrzeugen auf den Energieverbrauch und die Netzbelastung zu erstellen.

Die Ergebnisse der Simulationen bieten wertvolle Einblicke in die Wirtschaftlichkeit des Einsatzes von bidirektionalem Laden und zeigen, welche Auswirkungen diese Technologie auf den zukünftigen Netzausbau haben könnte. In der durchgeführten Langzeitsimulation wird aufgezeigt, dass bereits ab dem Jahr 2030 mit einem Anteil von ungefähr 30 Prozent an Elektrofahrzeugen (EVs) eine Reduzierung der Spitzenlast möglich ist. Bei der Analyse der gesamten Industrie führte das bidirektionale Laden zu einer Verringerung der kumulierten Spitzenlast um 282 MW im Hochlastzeitraum von 9 Uhr bis 10 Uhr.

Das abschließende Kapitel der Arbeit bewertet die gewonnenen Erkenntnisse kritisch, beleuchtet die Grenzen der Studie und gibt einen Ausblick auf zukünftige Forschungsfelder. Die Bachelorarbeit leistet damit einen wichtigen Beitrag zur aktuellen Debatte über die Rolle von Elektrofahrzeugen in der Energiewende und bietet Ansatzpunkte für weiterführende Forschungsarbeiten.

Abstract

This bachelor's thesis is dedicated to the analysis and evaluation of the use of bidirectional charging stations for electric vehicles at industrial sites.

Given the increasing share of renewable energies in the power grid and the rise of electric vehicles, this work investigates the potentials arising from the use of electric vehicle batteries as temporary energy storage. The focus is on the question of how this technology can contribute to optimizing energy consumption during peak load times and reducing peak demand.

The work follows a clear methodological approach, starting with the technical and theoretical foundations of electromobility, battery technology, and bidirectional charging. Based on this, a simulation model is developed that depicts various deployment scenarios at industrial sites. The modeling takes into account both technical and economic parameters and aims to create realistic forecasts about the impact of electric vehicles on energy consumption and grid load.

The results of the simulations provide valuable insights into the economic viability of using bidirectional charging and show the potential impacts of this technology on future grid expansion. The long-term simulation conducted demonstrated that as early as the year 2030, with approximately 30 percent of electric vehicles (EVs), a reduction in peak load is possible. When analyzing the entire industry, the implementation of bidirectional charging led to a reduction in the cumulative peak load by 282 MW.

The final chapter of the work critically evaluates the findings, highlights the limitations of the study, and provides an outlook on future research fields. Thus, the bachelor's thesis makes an important contribution to the current debate on the role of electric vehicles in the energy transition and offers starting points for further research.

Erklärungen	I
Danksagung	II
Kurzfassung/Abstract	III
Formelverzeichnis	IX
Tabellenverzeichnis	X
Abkürzungs- & Einheitenverzeichnis	XI
Einleitung.....	1
1 Forschungsstand.....	3
1.1 Forschungsbericht.....	4
1.1.1 Studie – Bidirektionales Laden in Deutschland – Marktentwicklung und Potenziale	4
1.1.2 BDL Bidirektionales Lademanagement - Abschlussbericht der FfE München.....	5
1.2 Datenerhebung der Simulation.....	6
2 Grundlagen und Theoretischer Rahmen	7
2.1 Mathematische Datenanalyse in Python.....	7
2.1.1 Berechnung des Mittelwertes	7
2.1.2 Normierung von Werten eines Datensatzes auf 1	8
2.1.3 Z-Score Normalisierung	8
2.1.4 Lineare Interpolation von Zwischenwerten.....	9
2.2 Spitzenlastreduktion	9
2.3 Elektrische Stromspeicher	10
3 Methodisches Vorgehen	12
3.1 Entwicklung der E-Mobilität	12
3.2 Verfügbarkeitsprofile von Elektrofahrzeugen.....	14
3.3 Entwicklung der Python-Ladesimulation	17
3.3.1 Schritt 1 - Zeitliche Auflösung und Interpolation des Industrie-Lastprofils.....	17
3.3.2 Schritt 2: Berechne Tageshöchstwerte.....	18
3.3.3 Schritt 3: Berechnung der EV – Verfügbarkeiten.....	19
3.3.4 Schritt 4: Berechnung der Batterie Zielzustände.....	22
3.3.5 Schritt 5: Berechnung der reellen Last und der täglichen Höchstlast	24
3.3.6 Schritt 6a: Algorithmus Simple Laden	25
3.3.7 Schritt 6b: Algorithmus Laden mit Ladebremse	26

3.3.8	Schritt 6c: Algorithmus Laden am Ende des Tages.....	29
3.3.9	Schritt 6d: Algorithmus Bidirektionales Laden	30
3.3.10	Schritt 7: Erstellung der Diagramme	32
3.3.11	Szenario Helpers Skript	36
4	Auswertung der Simulationsergebnisse.....	40
4.1	Simulation der gesamten Industrie Deutschlands.....	40
4.1.1	Kumulierte Leistungs- und Energiekennzahlen der E-Mobilität.....	41
4.1.2	Vergleich der verschiedenen Ladeszenarien	42
4.2	Ergebnisse der Chemie-Industrie.....	46
4.3	Ergebnisse der Automobileindustrie	48
4.4	Ergebnisse der Metall-Industrie.....	50
4.5	Ergebnisse Maschinenbau-Industrie.....	51
5	Diskussion und Ausblick	55
6	Fazit	57
7	Literaturverzeichnis.....	59
	Anhang	62

Abbildungsverzeichnis

Abbildung 1 - Lastspitzenglättung durch zeitliche Verschiebung des Energieverbrauchs (Eigene Darstellung)	10
Abbildung 2 - Anteile der Fahrzeugtypen ICE, PHEV und BEV von 2025 bis 2040 (Eigene Darstellung) [15]	13
Abbildung 3 - Anteil und Batteriekapazität nach Fahrzeugklassen im Elektromobilitätssektor (Eigene Darstellung) [16].....	13
Abbildung 4 - Prozentuale Verteilung des Arbeitsbeginns der Frühschicht (Eigene Darstellung)	15
Abbildung 5 - genormtes Industrielastprofil über eine Woche (Eigene Darstellung)	15
Abbildung 6 - Prozentuale Verteilung der Beschäftigten von Montag bis Sonntag (Eigene Darstellung)	16
Abbildung 7 - Zusammengefasste Beschäftigungsverteilung über eine Woche (Eigene Darstellung) .	17
Abbildung 8 – Schritt 1: Zeitliche Auflösung und Interpolation des Industrie-Lastprofils (Eigene Darstellung)	18
Abbildung 9 - Schritt 2: Tageshöchstwerte (Eigene Darstellung).....	19
Abbildung 10 - Schritt 3: Berechnung der BEV – Verfügbarkeiten - Teil 1 (Eigene Darstellung)	20
Abbildung 11 - Schritt 3: Berechnung der BEV – Verfügbarkeiten - Teil 2 (Eigene Darstellung)	21
Abbildung 12 - Schritt 4: Berechnung der Batterie Zielzustände Teil 1 (Eigene Darstellung).....	22
Abbildung 13 – Schritt 4: Berechnung der Batterie Zielzustände Teil 2 (Eigene Darstellung)	23
Abbildung 14 – Schritt 5: Berechnung der tatsächlichen Last und des täglichen Höchstwertes (Eigene Darstellung)	24
Abbildung 15 – Schritt 6a: Algorithmus Simple Laden (Eigene Darstellung)	25
Abbildung 16 -Schritt 6b: Algorithmus Laden mit Ladebremse (Eigene Darstellung).....	27
Abbildung 17 – Schritt 6c: Algorithmus Laden am Ende des Tages (Eigene Darstellung).....	29
Abbildung 18 – Schritt 6d: Algorithmus bidirektionales Laden (Eigene Darstellung)	31
Abbildung 19 – Schritt 7 Teil 1: Erstellung der Diagramme (Eigene Darstellung)	33
Abbildung 20 - Schritt 7 Teil 2: Erstellung der Diagramme (Eigene Darstellung).....	34
Abbildung 21 – Simulationsergebnisse des bidirektionalen Ladens in GW (Eigene Darstellung).....	35
Abbildung 22 – Simulationsergebnis Ladezustand SoC bidirektionales Laden in GWh (Eigene Darstellung)	35
Abbildung 23 – Vergleich original Lastprofil und Lastprofil mit bidirektionalem Laden in GW (Eigene Darstellung)	36
Abbildung 24 - Szenario Helpers Teil 1 (Eigene Darstellung)	37
Abbildung 25 - Szenario Helpers Teil 2 (Eigene Darstellung)	38
Abbildung 26 - Szenario Helpers Teil 3 (Eigene Darstellung)	39
Abbildung 27 – Energieverbrauch Strom in TWh/a der gesamten Industrie von 2030 bis 2045 (Eigene Darstellung) [20].....	40
Abbildung 28 - kumulierte Kapazitäts- und Leistungsentwicklung GWh/GW der Entwicklung der E-Mobilität von 2030 bis 2045(Eigene Darstellung)	41

Abbildung 29 - Lastprofil gesamte Industrie Szenario Simple Laden in GW 2045 (Eigene Darstellung)	42
Abbildung 30 - Lastprofil gesamte Industrie Szenario Ladebremse Laden in GW 2045 (Eigene Darstellung)	43
Abbildung 31 - Lastprofil gesamte Industrie Szenario „End of Day“ Laden in GW 2045 (Eigene Darstellung)	43
Abbildung 32 - Lastprofil gesamte Industrie Szenario „Bidi“ Laden in GW 2045 (Eigene Darstellung)	44
Abbildung 33 – Ladeenergie des bidirektionalen Ladens in GW 2045 (Eigene Darstellung)	45
Abbildung 34 – Szenario „Bidi“ Speicherkapazität SoC in GWh 2045 (Eigene Darstellung).....	45
Abbildung 35 - Prognostizierter Energieverbrauch der Chemie-Industrie in TWh/a bis 2045 (Eigene Darstellung) [20].....	46
Abbildung 36 - Lastprofil gesamte Chemie-Industrie Szenario Simple Laden in GW 2045 (Eigene Darstellung)	46
Abbildung 37 - Lastprofil gesamte Chemie-Industrie Szenario „Bidi Laden“ in GW 2045 (Eigene Darstellung)	47
Abbildung 38 - Szenario „Bidi“ Speicherkapazität SoC in GWh 2045 Chemie-Industrie (Eigene Darstellung)	48
Abbildung 39 - Prognostizierter Energieverbrauch der Automobil-Industrie bis 2045 (Eigene Darstellung) [20].....	48
Abbildung 40 - Lastprofil gesamte Automobil-Industrie Szenario „Simple“ 2045 (Eigene Darstellung)	49
Abbildung 41 - Lastprofil gesamte Automobil-Industrie Szenario „Bidi“ 2045 (Eigene Darstellung) ...	49
Abbildung 42 – Prognostizierter Energieverbrauch der Metall-Industrie bis 2045 (Eigene Darstellung) [20]	50
Abbildung 43 - Lastprofil gesamte Metall-Industrie Szenario „Bidi“ 2045 (Eigene Darstellung).....	51
Abbildung 44 - Szenario „Bidi“ Speicherkapazität SoC in GWh 2045 Metall-Industrie (Eigene Darstellung)	51
Abbildung 45 - Prognostizierter Energieverbrauch der Maschinenbau-Industrie bis 2045 (Eigene Darstellung) [20].....	52
Abbildung 46 - Lastprofil gesamte Maschinenbau-Industrie Szenario „Simple“ 2045 (Eigene Darstellung)	52
Abbildung 47 - Lastprofil Maschinenbau-Industrie Szenario Ladebremse Laden 2045 (Eigene Darstellung)	53
Abbildung 48 - Lastprofil gesamte Maschinenbau-Industrie Szenario „Bidi“ 2045 (Eigene Darstellung)	53

Formelverzeichnis

Formel 1 – Berechnung des Mittelwertes	7
Formel 2 - Normieren der Werte auf 1	8
Formel 3 - Berechnung der Standardabweichung	8
Formel 4 - Berechnung des Z-Score Normalisierung	9
Formel 5 – Lineare Interpolation	9

Tabellenverzeichnis

Tabelle 1 - Vergleich Anstieg der Spitzenlasten Gesamtindustrie 2030 bis 2045 (Eigene Darstellung)	44
Tabelle 2 - Vergleich Anstieg der Spitzenlasten Chemie-Industrie 2030 bis 2045 (Eigene Darstellung)	47
Tabelle 3 - Vergleich Anstieg der Spitzenlasten Automobile-Industrie 2030 bis 2045 (Eigene Darstellung)	50
Tabelle 4 - Vergleich Anstieg der Spitzenlasten Metall-Industrie 2030 bis 2045 (Eigene Darstellung)	51
Tabelle 5 – Vergleich Anstieg der Spitzenlasten Maschinenbau - Industrie 2030 bis 2045 (Eigene Darstellung)	54

Abkürzungs- & Einheitenverzeichnis

Einheit	Größe
GW	Gigawatt (1Mio. kW)
GWh	Gigawattstunde (1Mio. kWh)
KW	Kilowatt (1 Tsd. Watt)
KWh	Kilowattstunde (1 Tsd. Wattstunde)
MW	Megawatt (1 Tsd. kW)
MWh	Megawattstunde (1 Tsd. kW)
TWh	Terrawattstunde (1 Mrd. kWh)
Abkürzung	Beschreibung
AC	Wechselstrom
BDL	Bidirektionales Laden
CO ₂	Kohlenstoffdioxid
CSV	Comma-separated values
DC	Gleichstrom
EEG	Erneuerbare-Energien-Gesetz
EV	Elektro Vehicle
EFZ	Elektrofahrzeug
ICE	Fahrzeuge mit Verbrennungsmotoren (internal combustion engines)
LIS	Ladeinfrastruktur
PHEV	Plug-in-Hybride Elektro Vehikel
Pkw	Personenkraftwagen
PV	Photovoltaik
SoC	State of Charge
V2G	Vehicle to Grid
V2H	Vehicle to Home
V2L	Vehicle to Load
V2X	Vehicle to Everything
WindSeeG	Windenergie auf See Gesetz

Einleitung

Im Kontext des globalen Klimawandels und der wachsenden geopolitischen Spannungen im Zusammenhang mit fossilen Brennstoffen gewinnt die Frage nach nachhaltigen und unabhängigen Energiequellen immer mehr an Bedeutung. Mit der Novellierung des Klimaschutzgesetzes von 2019 hat die Bundesregierung sich das Ziel gesetzt, den Treibhausgas-Ausstoß bis 2030 um mindestens 65 Prozent gegenüber 1990 zu verringern und bis zum Jahr 2045 die Treibhausgasemission so weit zu mindern, dass eine Netto-Treibhausgasneutralität erreicht werden soll. [1] Im Energiesektor strebt die Bundesregierung einen Anteil Erneuerbarer Energien am Bruttostromverbrauch von 80 Prozent an. [1] Dies beinhaltet den beschleunigten Ausbau von Wind- und Solarenergie und den Ausstieg aus fossilen Energien. [2] Im Erneuerbaren Energie Gesetz EEG 2023 ist der Ausbaupfad bis 2045 für Solar- und Windenergie festgelegt. Laut EEG § 4 Ausbaupfad Absatz 1 und Absatz 3 sollen rund 115 Gigawatt (GW) Windenergie an Land und 215 GW Solarenergie erreicht werden. Außerdem ist laut §1 des WindSeeG Absatz 2 ein Ausbaziel für Windenergie auf See von mindestens 30 GW bis 2030 festgelegt. Der Ausstieg aus der Stromgewinnung durch Kohleverbrennung soll bis spätestens 2038 erfolgen, wobei angestrebt wird, diesen idealerweise bis 2030 vorzuziehen. [2] Die Bundesregierung hat rechtliche Grundlagen für einen weiteren Ausbau des Stromleitungsnetzes geschaffen, um den gesteigerten Anteil Erneuerbarer Energien und den wachsenden Strombedarf zu bewältigen. [3]

Es wird angestrebt, Verfahren für den Ausbau von Wind-, Solarparks und Stromnetzanbindungen deutlich zu beschleunigen. Der Ausstoß von Treibhausgasen im Mobilitätssektor soll bis 2030 um über 40 Prozent im Vergleich zu 2020 sinken, wofür die Elektromobilität gefördert wird. Die Bundesregierung hat das Ziel, bis 2030 mindestens 15 Millionen vollelektrische Pkw auf deutschen Straßen zu bringen. Bis 2030 sollen eine Million öffentlich und diskriminierungsfrei zugängliche Ladepunkte, insbesondere Schnellladestationen, verfügbar sein. [4] Diese Ziele sind entscheidend, um den Übergang zu einer nachhaltigeren und klimafreundlicheren Wirtschaft und Gesellschaft zu ebnen. Noch entscheidender aber sind smarte, technische Lösungen, um aus den hehren Zielen der Regierung, substantielle Effekte in der Wirklichkeit zu generieren, die sowohl ökologisch wie ökonomisch Sinn machen.

Daher widmet sich diese Untersuchung konkret der Forschungsfrage: Wie wirkt sich die Nutzung von Elektroauto-Batterien auf die Reduzierung der Spitzenlast in industriellen Energieversorgungssystemen aus? Dieses Thema ist besonders relevant, da es die Schnittstelle von Klimaschutz, Energieunabhängigkeit und der fortschreitenden technologischen Entwicklung berührt.

Die Abhängigkeit von Öl- und Gasimporten stellt für viele Länder ein signifikantes Risiko dar, sowohl in ökonomischer als auch in sicherheitspolitischer Hinsicht. Gleichzeitig erfordert die dringende Notwendigkeit des Klimaschutzes einen raschen Umstieg auf erneuerbare Energien und eine Abkehr von fossilen Brennstoffen. In diesem Zusammenhang bieten Elektrofahrzeuge nicht nur eine emissionsarme Mobilitätslösung, sondern ihre Batterien könnten auch als mobile Speicher für erneuerbare Energien dienen. Durch das bidirektionale Laden könnten EV-Batterien Energie in Zeiten

geringer Nachfrage speichern und in Zeiten hoher Nachfrage ins Netz zurückführen, was zu einer effizienteren Nutzung erneuerbarer Energien und zur Stabilisierung des Stromnetzes beiträgt.

Im Allgemeinen ist Schichtarbeit in der Industrie relativ verbreitet, insbesondere in Branchen, die eine kontinuierliche Produktion erfordern, wie die Automobilindustrie, Chemieindustrie, Stahlproduktion und Maschinenbauindustrie. Insofern können Schwarm Speicher von Elektrofahrzeugen (EVs) in 24 Stunden Schichtbetrieben eine kontinuierliche Reduzierung der Spitzenlast bereitstellen.

Das primäre Ziel dieser Arbeit ist die Untersuchung der Möglichkeiten und der ökonomischen Vorteile, die sich aus dem Einsatz von Batterien von Elektroautos zur Stabilisierung des Stromnetzes und zur Minderung der Spitzenlast in industriellen Bereichen ergeben. Ein zentraler Aspekt der Untersuchung ist die Betrachtung der Entwicklung der Elektrifizierung des Verkehrssektors und Industrie bis zum Jahr 2045. Speziell für den Industriesektor wird das Potenzial der Ladeinfrastruktur (LIS) an Produktionsstandorten bewertet, wobei sowohl der ökonomische als auch der netztechnische Nutzen für die Anwender hervorgehoben werden. Die Ergebnisse dieser Arbeit werden durch eine Simulation in Python programmiert.

Den Ausgangspunkt bildet ein Überblick über die Grundlagen der Elektromobilität und der bidirektionalen Ladetechnik. Anschließend wird im methodischen Teil auf die Simulationsparameter und die Python Programmierung eingegangen, gefolgt von einer detaillierten Analyse der Potentiale des BDL dieses Ansatzes. Daraufhin werden die Auswirkungen auf die Spitzenlastreduzierung untersucht. Die Arbeit schließt mit einer Diskussion der Ergebnisse und einem Fazit ab.

1 Forschungsstand

Technisch betrachtet bedeutet bidirektionales Laden nichts anderes als die Rückspeisefähigkeit der Batteriespeicher von Elektrofahrzeugen in das Stromnetz. Hierbei werden die geladenen Batteriespeicher der EVs entladen, um Strom bspw. bei Netzengpässen in das Netz zurückzuführen. Das Laden der EVs umfasst mehrere technische Schritte, die die Übertragung von elektrischer Energie aus einer externen Quelle in die Batterie der EVs ermöglichen. Elektrofahrzeuge können an verschiedene Arten von Ladestationen angeschlossen werden, von einfachen Haushaltssteckdosen (langsames Laden) bis hin zu speziellen Schnellladestationen. Bei den meisten öffentlichen und Heimpladestationen wird Wechselstrom (AC) bereitgestellt. Die meisten Elektrofahrzeuge verfügen über einen eingebauten Wechselrichter, der AC in Gleichstrom (DC) umwandelt, um die Batterie zu laden. Bei Schnellladestationen wird direkt Gleichstrom geliefert, der die eingebauten Wechselrichter des Fahrzeugs umgeht und eine schnellere Aufladung ermöglicht. [5]

Die Rückspeisefähigkeit von EVs in ein externes elektrisches System oder Netzwerk wird als Vehicle-to-Everything (V2X) bezeichnet. Neben der Interaktion mit dem Stromnetz (V2G) kann dies auch die Energieübertragung zu Wohngebäuden (V2H - Vehicle-to-Home), Bürogebäuden (V2B - Vehicle-to-Building) oder anderen elektrischen Systemen einschließen. [6] V2X-Systeme sind vielseitiger, da sie nicht nur auf die Interaktion mit dem Stromnetz beschränkt sind, sondern auch private, kommerzielle oder industrielle Anwendungen unterstützen können. Zusammenfassend lässt sich sagen, dass V2G sich speziell auf die Interaktion zwischen Elektrofahrzeugen und dem Stromnetz konzentriert, während V2X ein breiterer Begriff ist, der jede Art von Energieaustausch zwischen einem EFZ und externen Systemen oder Netzwerken umfasst. Beide Konzepte erfordern fortgeschrittene Technologien in Bezug auf Hardware, Kommunikation und Steuerung, um eine effiziente und effektive Nutzung der in EV-Batterien gespeicherten Energie zu ermöglichen.

Verschiedene Fahrzeughersteller haben begonnen, Vehicle-to-Everything (V2X) Technologien in ihren Fahrzeugen zu integrieren. Einer der Pioniere im Bereich V2G mit Fahrzeugen ist der Nissan Leaf, der die Technologie unterstützt, um Energie zwischen dem Fahrzeug und dem Stromnetz auszutauschen. [7] Mitsubishi bietet ebenfalls V2G-fähige Fahrzeuge an, insbesondere durch Modelle wie den Mitsubishi Outlander / iMIEV, der V2H-Funktionalitäten unterstützt. [8] Tesla hat Interesse an V2G-Technologien gezeigt, obwohl das Unternehmen sich bisher hauptsächlich auf Vehicle-to-Load (V2L) und Vehicle-to-Home-Anwendungen konzentriert hat. Tesla-Autos sind bekannt für ihre fortschrittlichen Batterietechnologien und das Potenzial für zukünftige V2X-Integrationen. [9] BMW hat in einigen Pilotprojekten mit V2G-Technologien experimentiert. Der BMW i3 wurde in verschiedenen Studien als Teil von V2G-Experimenten eingesetzt. [10] Hyundai hat in neueren Modellen wie dem Hyundai Ioniq 5 und dem Hyundai Kona Electric V2L-Funktionen integriert, die zeigen, dass das Unternehmen die V2X-Technologie weiter erforscht und entwickelt. [8] Honda hat mit dem Honda Power Manager Concept ebenfalls Interesse an V2G-Technologien gezeigt und experimentiert mit Möglichkeiten, Fahrzeuge in das Heimenergiesystem zu integrieren. [11]

Für das Lademanagement von Ladeinfrastrukturen, speziell für Elektrofahrzeuge, gibt es bereits verschiedene technische Lösungen, die darauf abzielen, die Effizienz und Effektivität des Ladevorgangs zu optimieren. Diese Technologien sind entscheidend, um eine nahtlose Integration von Elektrofahrzeugen in das Stromnetz zu gewährleisten, insbesondere im Hinblick auf bidirektionales Laden und Netzstabilisierung. Die Smart Charging Technologie ermöglicht es, den Ladevorgang basierend auf verschiedenen Faktoren wie dem Zustand des Stromnetzes, den Strompreisen und dem tatsächlichen Energiebedarf des Fahrzeugs zu steuern. Smart Charging-Systeme können auch die Ladevorgänge so planen, dass sie in Zeiten niedrigerer Strompreise oder geringerer Netzbelastung stattfinden. [12] Demand Response Management Systeme ermöglichen eine dynamische Anpassung der Ladevorgänge an die Bedürfnisse des Stromnetzes. Beispielsweise können Elektrofahrzeuge während Zeiten hoher Netzbelastung weniger oder langsamer geladen werden, um das Netz zu entlasten. [13] Einige Ladestationen sind mit eigenen Batteriespeichersystemen ausgestattet, um Energie zu Zeiten geringer Nachfrage zu speichern und sie während Spitzenlastzeiten zur Verfügung zu stellen. Dies hilft, die Belastung des Netzes zu verringern. Bei einem aktuellen Projekt am Flughafen München entstehen hunderte Elektroladestationen, die in Kombination mit 2 PV-Anlagen und einem 1 Megawattstunden (MWh) Energiespeicher gekoppelt werden. [14] Um eine breite Akzeptanz und Effizienz zu gewährleisten, arbeiten viele Lademanagementsysteme gemäß internationalen Standards und Protokollen, die eine Interoperabilität zwischen verschiedenen Fahrzeugen und Ladestationen ermöglichen. Viele Lademanagementsysteme bieten Benutzerschnittstellen über mobile Apps, die es den Nutzern ermöglichen, ihre Ladevorgänge zu überwachen, zu steuern und zu planen.

Diese Technologien sind entscheidend für die effiziente Integration von Elektrofahrzeugen in das Stromnetz und spielen eine wichtige Rolle bei der Maximierung der Wirtschaftlichkeit und Umweltverträglichkeit der Elektromobilität. Sie tragen dazu bei, die Herausforderungen im Zusammenhang mit Spitzenlasten und der Netzstabilität zu bewältigen.

1.1 Forschungsbericht

Die erforderliche Technologie ist bereits seit mehreren Jahren bekannt und vorhanden, wird aber noch nicht in voller Breite in EVs integriert, da das Potential für die erweiterte Speichernutzung noch zu wenig an Bedeutung erlangt hat. Aktuell sind verschiedene Akteure aus Wissenschaft, Industrie und Forschungseinrichtungen im Bereich des bidirektionalen Ladens aktiv. Im Folgenden werden die Ergebnisse der aktuellen Studien zum Thema BDL zusammengefasst.

1.1.1 Studie – Bidirektionales Laden in Deutschland – Marktentwicklung und Potenziale

Die Studie „Bidirektionales Laden in Deutschland – Marktentwicklung und Potenziale“ von NRW.Energy4Climate und e-mobil BW 2023 hebt hervor, dass das bidirektionale Laden von Elektroautos großes Potenzial für die Sektorenkopplung birgt. Es ermöglicht, Elektrofahrzeuge als mobile Zwischenspeicher für Strom zu nutzen, wodurch die Stromnetze entlastet und große Mengen Erneuerbarer Energie zwischengespeichert werden könnten. Dies kann zur Netzstabilisierung beitragen und als zukünftiges Geschäftsmodell für Netzbetreiber und Nutzer dienen. [15]

Die Potenzialermittlung in dieser Studie beruht auf marktspezifischen Faktoren, die durch gesetzliche und regulatorische Vorgaben, sowie herstellereinspezifische Elektrifizierungs- und CO₂ – Flottenziele prognostiziert wurden. Hierzu wurden drei verschiedene Szenarien untersucht. Im ersten Szenario wird die Fahrzeugbatterie in das lokale Stromnetz integriert, das idealerweise mit einer PV-Anlage kombiniert betrieben wird. Im zweiten Szenario wird die Arbitrage im System V2G betrachtet und im dritten Szenario die Netzdienstleistung in Bezug auf der Sicherstellung der Energieversorgung.

1.1.2 BDL Bidirektionales Lademanagement - Abschlussbericht der FfE München

Das Forschungsprojekt "Bidirektionales Lademanagement (BDL)" zielt darauf ab, ein ganzheitliches, nutzerorientiertes Angebot für die Integration von bidirektionalen Elektrofahrzeugen in das Energiesystem zu entwickeln und zu erproben. Dieses intelligente Zusammenspiel von Elektrofahrzeugen, Ladeinfrastruktur und Energiesystem soll sowohl die Netzstabilität als auch die Versorgungssicherheit erhöhen und gleichzeitig eine maximale Nutzung erneuerbarer Energien ermöglichen. Ein Pilotbetrieb mit 50 rückspeisefähigen BMW i3s diente dazu, den Kundennutzen und die Wertigkeit der Technologie in verschiedenen Anwendungsfällen zu demonstrieren. [16]

Die Forschungsstelle für Energiewirtschaft (FfE) konzentrierte sich im Rahmen des BDL-Projekts auf verschiedene Themenbereiche, darunter die Entwicklung von Use Cases für bidirektionale Elektrofahrzeuge, die Untersuchung von Erlöspotenzialen und Kosten dieser Use Cases, die Bewertung regulatorischer Rahmenbedingungen, die Analyse von Netzbelastungen durch bidirektionale Elektrofahrzeuge im Verteilnetz, die Auswirkungen auf das Energiesystem und die Erarbeitung eines Messkonzepts sowie die Datenauswertung des Feldversuchs. [16]

Für die verschiedenen definierten Use Cases wurden Mehrwerte aus Sicht der Akteure diskutiert und ausgewertet, einschließlich Erlösmöglichkeiten, Möglichkeiten zur Verringerung von Treibhausgas-Emissionen und die Erhöhung des Autarkiegrads eines Haushalts (beispielsweise durch PV-Eigenverbrauchsoptimierung). Für jeden bewerteten Use Case wurde ein prägnanter Steckbrief erstellt, der die Mehrwerte des Use Cases zusammenfasst. Zusätzlich wurden die Mehrkosten des Gesamtsystems für bidirektionales Laden bewertet, um die Wirtschaftlichkeit der Use Cases zu berechnen. [16]

Ein weiterer wichtiger Aspekt des Projekts war die Adressierung regulatorischer Hürden, die einer Erschließung der Flexibilität von bidirektionalen Elektrofahrzeugen entgegenstehen könnten. Diese Themen wurden in Fachgremien und Arbeitsgruppen integriert, und das Konsortium verfasste Positionspapiere zu verschiedenen Use Case-Gruppen. [16]

Die FfE führte auch eine Analyse der zukünftigen Netzbelastungen durch bidirektionale Elektrofahrzeuge in verschiedenen Use Cases durch. Hierbei zeigte sich, dass bidirektionales Laden die Netzbelastung je nach Use Case unterschiedlich beeinflussen kann, wobei die Auswirkungen auf die Netzbelastung variieren können, je nachdem ob die Ladevorgänge auf Eigenverbrauchserhöhung oder rein tarifliche Optimierung anhand variabler Preise ausgerichtet sind. [16]

Aus systemischer Sicht führt die Integration bidirektionaler Elektrofahrzeuge in die Strommärkte zu einer Erhöhung der Liquidität am Strommarkt und somit zu volkswirtschaftlichen Mehrwerten. Es

wurde eine optimale Durchdringungsrate von bidirektionalen Elektrofahrzeugen im europäischen Energiesystem bestimmt, wobei diese Integration auch Auswirkungen auf andere Elemente des Energiesystems hat, wie eine bessere Integration von PV-Energie und eine Reduzierung des Bedarfs an Gas- und Wasserstoffkraftwerken zur Deckung der Spitzenlasten. [16]

1.2 Datenerhebung der Simulation

Für die Durchführung der Berechnungen und Simulationen dieser Arbeit wurden Daten aus diversen Quellen herangezogen. Die Verteilung der Beschäftigungszahlen basiert auf den Erhebungen des Wirtschafts- und Sozialwissenschaftlichen Instituts (WSI) sowie des Instituts für Arbeitsmarkt- und Berufsforschung (IAB). [17] [18] Die reellen Zahlen der Beschäftigung der Industriezweige wurden vom Bundesministerium für Wirtschaft und Klimaschutz erhoben. [19] Die kumulierten Lastprofile der Industrie wurden mithilfe der Daten der Forschungsstelle für Energiewirtschaft (FfE) München erarbeitet (Publication source: Ganz, Kirstin et. al., Forschungsgesellschaft für Energiewirtschaft mbH (FfE)). Die erfassten Energieverbräuche der verschiedenen Industriezweige stützen sich auf Angaben der Arbeitsgemeinschaft Energiebilanzen e.V. [20] Für die Programmierung der Simulationen wurde vornehmlich auf die Bibliotheken Pandas (pandas.pydata.org) und Numpy (numpy.org) in Python (www.python.org) zurückgegriffen.

2 Grundlagen und Theoretischer Rahmen

Die Nutzung von Python für die Datenanalyse, das Management von Energieflüssen, insbesondere durch bidirektionales Laden und Spitzenlastmanagement in Verbindung mit Energiespeichern, bildet ein innovatives Feld in der modernen Energietechnik. Diese Themenbereiche sind nicht nur für die Entwicklung nachhaltiger Energieinfrastrukturen entscheidend, sondern auch für die Optimierung bestehender Systeme hinsichtlich Effizienz und Zuverlässigkeit.

2.1 Mathematische Datenanalyse in Python

Python ist eine vielseitige, interpretierte Programmiersprache, die sich durch ihre einfache Syntax und Lesbarkeit auszeichnet, was sie zu einer beliebten Wahl für Anfänger und Experten gleichermaßen macht. Sie wurde in den späten 1980er Jahren von Guido van Rossum entwickelt und ist heute eine der meistgenutzten Sprachen für eine Vielzahl von Anwendungen, von Webentwicklung über wissenschaftliche und numerische Berechnungen bis hin zu maschinellem Lernen und Datenanalyse.

Die Anwendung von Python in der Mathematik und Datenanalyse basiert auf grundlegenden mathematischen Prinzipien und Methoden. Ein fundiertes Verständnis dieser Grundlagen vereinfacht das Nachvollziehen mathematischer Berechnungen in Python und den effektiven Einsatz der umfangreichen Bibliotheken und Werkzeuge, die Python zur Verfügung stehen. Im Folgenden wird ein Überblick über die zentralen mathematischen Kenntnisse gegeben, die für den Einsatz von Python erforderlich sind.

2.1.1 Berechnung des Mittelwertes

Der Mittelwert ist ein fundamentales Maß der zentralen Tendenz in der Statistik und wird verwendet, um die durchschnittliche Größe einer Datenmenge zu beschreiben. Die Formel zur Berechnung des Mittelwertes einer Reihe von n Werten lautet: [21]

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

Formel 1 – Berechnung des Mittelwertes

Hierbei ist:

\bar{x} der Mittelwert der Datenreihe,

n die Anzahl der Werte in der Datenreihe, und

x_i der i -te Wert in der Datenreihe.

Der Mittelwert wird berechnet, indem die Summe aller Werte in der Datenreihe gebildet und dann durch die Anzahl dieser Werte geteilt wird. Diese Berechnung liefert einen einzigen Wert, der die durchschnittliche Größe der Werte in der Datenreihe repräsentiert.

Um den Mittelwert einer Pandas Series oder eines DataFrame in Python zu berechnen, wird die `mean()` Methode auf dem entsprechenden Objekt angewandt [zum Bsp. `mittelwert = data.mean()`]. Die

Funktion kann sowohl auf die ganze Serie/DataFrame angewendet werden als auch auf spezifische Achsen innerhalb eines DataFrame (Zeilen oder Spalten).

2.1.2 Normierung von Werten eines Datensatzes auf 1

Um die Werte in einem Datensatz so zu normieren, dass die Summe ihrer Absolutbeträge gleich 1 ergibt, verwendet man die folgende Formel für die Normalisierung jedes Wertes x_i m Datensatz.

Die Formel 2 zur Berechnung lautet:

$$NW_i = \frac{x_i}{\sum_{j=1}^n |x_j|}$$

Formel 2 - Normieren der Werte auf 1

Hierbei ist:

NW_i der Normierte $Wert_i$

x_i der i -te Wert im Datensatz,

$|x_j|$ der Absolutbetrag des j -ten Wertes im Datensatz, und

n die Gesamtanzahl der Werte im Datensatz.

Die Summe im Nenner $\sum_{j=1}^n |x_j|$, addiert die Absolutbeträge aller Werte im Datensatz. Indem man jeden Wert durch diese Summe teilt, erhält man den normierten $Wert_i$, der so skaliert ist, dass die Summe der Absolutbeträge aller normierten Werte gleich 1 ist. [22]

Diese Methode der Normalisierung ist besonders nützlich, wenn die relativen Anteile oder Proportionen der Werte im Datensatz von Interesse sind, unabhängig von deren Vorzeichen. Sie wird häufig in Anwendungen verwendet, bei denen die Verteilung der relativen "Gewichte" der einzelnen Werte wichtig ist, wie z.B. bei der Erstellung von Lastprofilen in der numerischen Optimierung oder bei der Verarbeitung von Wahrscheinlichkeitsverteilungen.

2.1.3 Z-Score Normalisierung

Bei der Z-Score-Normalisierung ist die Standardabweichung ein Maß für die Streuung oder Variabilität der Datenpunkte in einem Datensatz relativ zum Mittelwert der Datenreihe. Die Standardabweichung gibt an, wie weit sich die einzelnen Werte im Durchschnitt vom Mittelwert entfernen [22].

Die Formel 3 zur Berechnung der Standardabweichung für eine Stichprobe lautet:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \mu)^2}{n - 1}}$$

Formel 3 - Berechnung der Standardabweichung

Der Z-Score eines Datenpunkts wird berechnet, indem der Mittelwert von diesem Datenpunkt subtrahiert und das Ergebnis durch die Standardabweichung der Datenreihe geteilt wird:

$$\sigma = \frac{x_i - \mu}{\sigma}$$

Formel 4 - Berechnung des Z-Score Normalisierung

Dabei ist:

σ die Standardabweichung,

n die Anzahl der Datenpunkte,

x_i der einzelne Datenpunkt, und

μ der Mittelwert (Durchschnitt) der Datenreihe.

2.1.4 Lineare Interpolation von Zwischenwerten

Die lineare Interpolation ist eine Methode, um einen unbekanntem Wert auf einer Geraden zwischen zwei bekannten Werten zu schätzen. Die grundlegende Formel der linearen Interpolation kann ausgedrückt werden als:

$$y = y_1 + \frac{(x - x_1) * (y_2 - y_1)}{x_2 - x_1}$$

Formel 5 – Lineare Interpolation

Hierbei ist:

y : der geschätzte Wert an der Stelle x ,

x : der Punkt, an dem wir den Wert schätzen wollen,

x_1 und x_2 : die x - Werte der beiden bekannten Punkte, zwischen denen wir interpolieren wobei $x_1 < x_2$,

y_1 und y_2 : die y -Werte der beiden bekannten Punkte, die zu x_1 und x_2 gehören.

Die Funktion `np.interp()` in Python wird durch den Import der Bibliothek Numpy verfügbar. Diese Formel basiert auf der Annahme, dass sich der Wert linear zwischen x_1, y_1 und x_2, y_2 ändert. Das bedeutet, dass die Rate der Veränderung zwischen diesen zwei Punkten konstant ist. Die Formel berechnet im Wesentlichen die Steigung der Geraden, die durch die Punkte (x_1, y_1) und (x_2, y_2) geht, und verwendet diese Steigung, um den y -Wert an der gewünschten Stelle x zu schätzen. [23]

2.2 Spitzenlastreduktion

Für viele Industriebetriebe spielen neben dem Energieverbrauch auch die höchsten Lastspitzen innerhalb eines Jahres eine wesentliche Rolle bei der Berechnung der Stromkosten. Gemäß § 19 Abs. 1 der Stromnetzentgeltverordnung (StromNEV) erfolgt jedoch eine Abrechnung auf Basis der monatlichen Lastspitzen, sofern in bestimmten Monaten ein signifikant gesteigener Verbrauch zu verzeichnen ist. Der Leistungspreis, der vor allem für Verbraucher mit einem Jahresverbrauch von über 100 MWh nach § 17 Abs. 6 StromNEV relevant ist, bildet zusammen mit dem Grund-, Arbeits- und Blindarbeitspreis die Grundlage für die Netzentgelte. Die Verringerung von Lastspitzen, bekannt unter Begriffen wie Lastspitzenkappung oder Spitzenlastreduzierung, führt daher direkt zu einer Senkung der Kosten. [24]

Die Minimierung unerwarteter Lastspitzen auf der Verbraucherseite bringt signifikante Vorteile für das vorangestellte Energiesystem mit sich, da sie eine Entlastung der Stromnetze bewirkt. Da die Netzinfrastruktur auf den maximalen Lastfall ausgelegt ist, kann durch ein intelligentes Lastmanagement der Bedarf für Netzausbaumaßnahmen reduziert werden. Darüber hinaus führen geringere Lasten zu einer Verringerung der Übertragungsverluste. Durch eine gleichmäßigere Verteilung des Strombedarfs verringert sich der Einsatz teurer und ineffizienter Spitzenlastkraftwerke, was wiederum einen wesentlichen Beitrag zur Erreichung klimapolitischer Ziele leistet. [24]

Zur Reduktion von Lastspitzen könnte theoretisch eine Abschaltung von Produktionsanlagen in Betracht gezogen werden. Dies würde jedoch einen tiefgreifenden Eingriff in die sensiblen Produktionsprozesse darstellen und ist daher zu vermeiden. Stattdessen wird der Einsatz von Batteriespeichern bevorzugt, die in Momenten drohender Lastspitzen entladen und während Phasen geringerer Last wieder aufgeladen werden, wie in Abbildung 1 ersichtlich. Diese Strategie ermöglicht eine effektive Glättung der Lastprofile, ohne die Produktionskontinuität zu gefährden. [24]

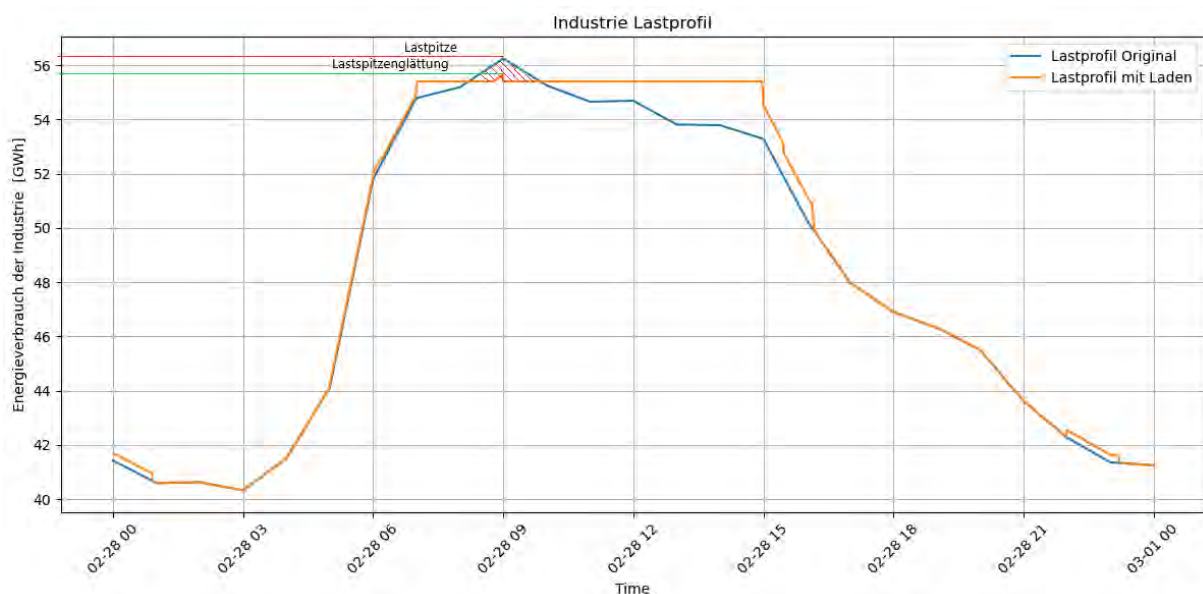


Abbildung 1 - Lastspitzenglättung durch zeitliche Verschiebung des Energieverbrauchs (Eigene Darstellung)

2.3 Elektrische Stromspeicher

Ein Energiespeichersystem kann man als eine Vorrichtung definieren, die durch Aufnahme von Energie oder eines Energieträgers ihren Energiegehalt erhöht (dieser Vorgang wird als Laden bezeichnet). Diese Energie wird anschließend über eine gewisse Zeit mit möglichst geringen Verlusten konserviert, um bei Notwendigkeit wieder freigesetzt zu werden (Entladen), was eine Verringerung des gespeicherten Energiegehalts zur Folge hat. Jedoch gehen mit der Nutzung eines solchen Speichersystems unvermeidliche Energieverluste einher, etwa durch Standverluste oder während der Lade- und Entladezyklen. Des Weiteren fallen zusätzliche Investitions- und Betriebskosten für das System an, ohne dass dadurch die Leistungsfähigkeit von Energieerzeugungsanlagen direkt substituiert wird. [25]

Während elektrische Netze dafür sorgen, dass räumliche Differenzen zwischen Energieerzeugung und -verbrauch ausgeglichen werden, dienen Energiespeicher dazu, zeitliche Diskrepanzen zwischen Erzeugung und Verbrauch zu überbrücken, sodass Erzeugung und Verbrauch voneinander entkoppelt werden können. Überschüssiger Strom kann beispielsweise zwischengespeichert und zu einem späteren Zeitpunkt genutzt werden, um Schwankungen in der Energieerzeugung (als Primär- und Sekundärreserve) oder bei Energiemangel (etwa während wind- und sonnenarmer Perioden) zu kompensieren. Zudem sind Energiespeicher für die unterbrechungsfreie Stromversorgung von entscheidender Bedeutung, etwa bei der Notstromversorgung von Krankenhäusern und Rechenzentren oder zur Versorgungssicherheit bei Stromausfällen. [25]

3 Methodisches Vorgehen

Um die Fragestellung „wie sich die Nutzung von Elektroauto-Batterien zur Netzstabilisierung und Spitzenlastreduzierung auswirkt“ zu beantworten, wird ein mehrdimensionaler Ansatz gewählt. Dieser umfasst eine grundlegende Literaturrecherche zur Elektromobilität und deren Ladeinfrastruktur, sowie der Beschäftigungsverteilung und den Energieverbräuchen der Industrien. Die Kernkomponente dieser Untersuchung bildet eine umfassende Simulation, die darauf abzielt, ein detailliertes Verständnis der Auswirkungen und Potenziale der Nutzung von Elektrofahrzeugen (EVs) mit bidirektionalen Lademöglichkeiten als temporäre Energiespeicher in Industriestandorten zu erlangen. Die Simulation wird entwickelt, um eine Vielzahl von Zukunftsszenarien zu modellieren, die sich aus der zunehmenden Elektrifizierung der Mobilitäts- und Industriebereiche ergeben. Konkret werden Szenarien für den Zeitraum von 2030 bis 2045 betrachtet, wobei besonderes Augenmerk auf die Entwicklung der Elektromobilität und die fortschreitende Elektrifizierung industrieller Prozesse gelegt wird.

Durch die Analyse dieser Szenarien soll die Simulation aufzeigen, wie sich unterschiedliche Entwicklungspfade der Elektromobilität und der industriellen Elektrifizierung auf den Energieverbrauch und die Stabilität des Stromnetzes auswirken. Zentral ist hierbei die Frage, inwieweit EVs mit der Fähigkeit zum bidirektionalen Laden als Zwischenspeicher agieren können, um nicht nur die Netzbelastung insgesamt zu minimieren, sondern insbesondere auch den Bedarf an Netzstrom während der Spitzenlastzeiten zu reduzieren. Dabei sollen Effekte wie Lastverschiebung und Lastglättung durch die intelligente Steuerung der Ladeprozesse und die Rückeinspeisung gespeicherter Energie ins Netz detailliert untersucht werden.

Diese methodische Herangehensweise ermöglicht es, präzise Einsichten der wirtschaftlichen und ökologischen Vorteile zu gewinnen, die sich durch den Einsatz von EVs als temporäre Energiespeicher ergeben können. Darüber hinaus trägt die Simulation dazu bei, das Potenzial zur Reduzierung des Bedarfs an zusätzlichem Netzausbau zu quantifizieren und somit die Effizienz des Gesamtsystems zu bewerten. Die Ergebnisse dieser Simulation sollen einen Beitrag zur Diskussion um die Rolle der Elektromobilität in der Energiewende und insbesondere zur Stabilisierung des Stromnetzes leisten.

3.1 Entwicklung der E-Mobilität

Im ersten Schritt werden anhand der Literaturergebnisse die Entwicklungen des Mobilitätssektors bis 2045 bewertet, wie in Abbildung 2 Anteile der Fahrzeugtypen mit Verbrennungsmotoren (ICE), Plug-in-Hybride (PHEV) und vollelektrische Batteriefahrzeuge (BEV) dargestellt, um somit die Anforderungen für die Ladeinfrastrukturen (LIS) zu bestimmen. Aufgrund der rasanten Entwicklung von autonomen Fahrzeugen und dem bisherigen Ausbau der öffentlichen Verkehrsinfrastruktur, wird dieser Teil der Simulation variabel gestaltet und als Best-Case und Worst-Case Szenario untersucht.

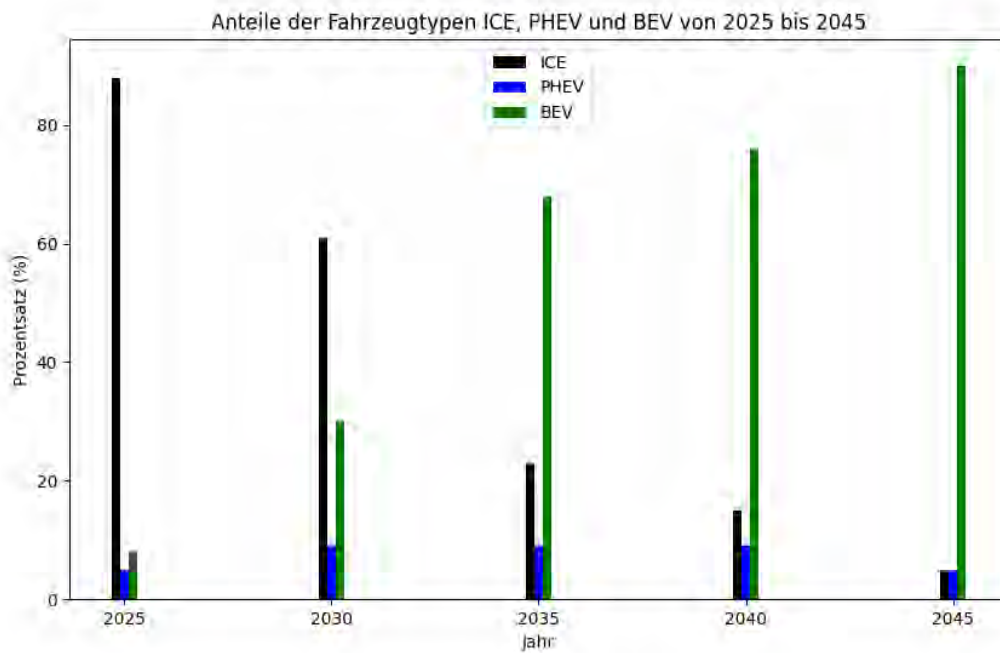


Abbildung 2 - Anteile der Fahrzeugtypen ICE, PHEV und BEV von 2025 bis 2040 (Eigene Darstellung) [15]

Ferner müssen die Anteile von Kleinwagen, Mittelklasse und Oberklasse ermittelt werden, da diese unterschiedlichen Speicherkapazitäten besitzen und somit das BDL entsprechend beeinflussen. In Abbildung 3 sind die Anteile der Fahrzeugklassen des Kraftfahrtbundesamtes kumuliert zusammengefasst und die gemittelten Batteriekapazitäten der entsprechenden Klassen dargestellt. [16]

Anteil und Batteriekapazität nach Fahrzeugklassen im Elektromobilitätssektor

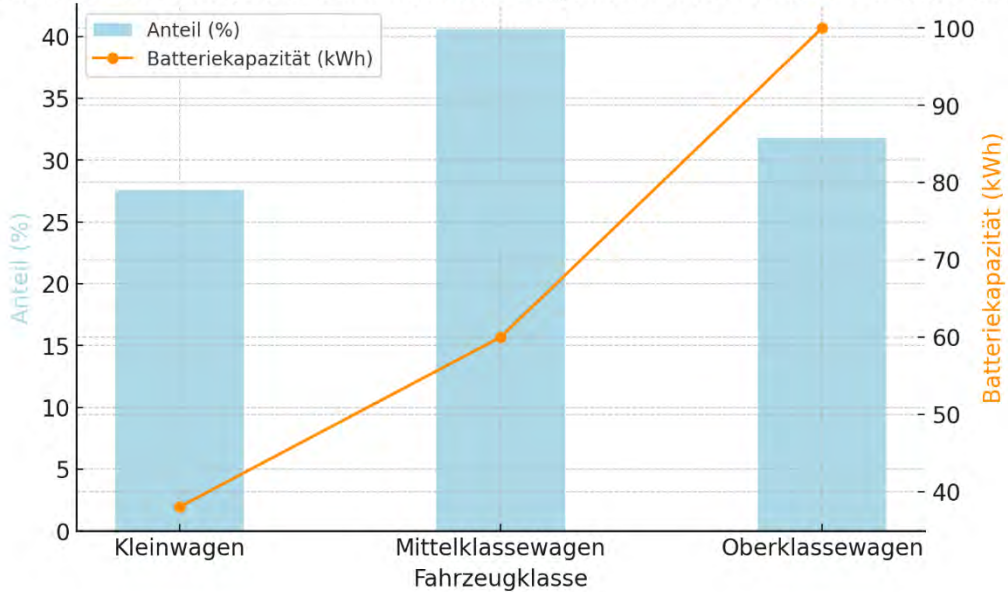


Abbildung 3 - Anteil und Batteriekapazität nach Fahrzeugklassen im Elektromobilitätssektor (Eigene Darstellung) [16]

3.2 Verfügbarkeitsprofile von Elektrofahrzeugen

Um die Dynamik und die potenzielle Rolle von EVs als mobile Energiespeicher im Rahmen des BDL adäquat zu erfassen, erscheint die Entwicklung eines umfassenden Modells als essenziell. Dieses Modell soll in der Lage sein, die Verfügbarkeit von EVs für BDL-Aktivitäten zu prognostizieren, indem es unterschiedliche Arbeitszeitregime und die Verteilung der Beschäftigungsverhältnisse über die Woche hinweg berücksichtigt. Die Modellierung muss dabei verschiedene Beschäftigungstypologien wie Vollzeitarbeit, Teilzeitarbeit, geringfügige Beschäftigungsverhältnisse und Schichtarbeit integrieren, um die Variabilität der Fahrzeugverfügbarkeit für das bidirektionale Laden zu reflektieren. Diese methodische Herangehensweise ermöglicht es, die zeitliche und potenzielle Kapazität von EVs als dezentrale Energiespeicher im Stromnetz präziser zu bestimmen und somit deren Beitrag zur Netzstabilisierung und zur Reduktion von Spitzenlasten fundiert zu evaluieren.

Basierend auf den Daten des Wirtschafts- und Sozialwissenschaftlichen Instituts (WSI) und den Daten des Instituts für Arbeitsmarkt- und Berufsforschung (IAB) wird ein Profil zur Verteilung der Beschäftigung in der Industrie erstellt.

In der Industrie beträgt der Anteil der Vollzeitbeschäftigung 88 %, jener der Teilzeitbeschäftigung 7 % und der Anteil der geringfügigen Beschäftigung 5 %. [17] Für die Simulation wird lediglich der Anteil der Vollzeitbeschäftigung berücksichtigt, da diese den größten Anteil der Beschäftigten darstellt und für das BDL die Speicherverfügbarkeit von Teilzeit- und geringfügig Beschäftigten nicht relevant erscheint. Zudem wurde bei der Simulation mit Teilzeit- und geringfügig Beschäftigten die Berechnungszeit nahezu vervierfacht und erzielte keine nennenswerten Unterschiede der Ergebnisse.

Für die Berechnung zur Verteilung der Vollzeitbeschäftigung muss berücksichtigt werden, dass der Anteil der Beschäftigung an Samstagen 17,5 % und an Sonntagen, sowie an Feiertagen ca. 9 % beträgt. Der Anteil zu welcher Tageszeit Vollzeitbeschäftigte arbeiten gehen, gliedert sich in drei Tagesschichten, wobei der überwiegende Anteil der Beschäftigten 81,5 % zwischen 5 Uhr und 18 Uhr arbeiten. Dieser Anteil inkludiert auch die Beschäftigten, die tagsüber in Schicht arbeiten. Der Anteil der Beschäftigten, die im Schichtdienst zwischen 14 Uhr und 22 Uhr arbeiten, beträgt 15,2 % und der geringste Anteil von 4,65 % arbeitet zwischen 22 Uhr und 6 Uhr. [18]

Obendrein muss berücksichtigt werden, dass der Anteil der Beschäftigten, die zwischen 5 Uhr und 18 Uhr arbeiten, zu unterschiedlichen Zeiten ihre Arbeit beginnen. Der IAB-Forschungsbericht spiegelt den Arbeitsmarkt aller Beschäftigungsgruppen in Deutschland wider, daher muss der Anteil der Schichtarbeit in der Industrie mit den Daten des WSI neu gewichtet werden. Wie in Abbildung 4 dargestellt, verteilt sich der Arbeitsbeginn der Tagesschicht zwischen 5 Uhr und 10 Uhr. Die Werte ab 11 Uhr bis 4 Uhr morgens liegen bei einem Prozent und fließen daher nicht in die Berechnung mit ein. Zusätzlich wurde der Anteil der Beschäftigten berücksichtigt, die mit dem Auto zur Arbeit fahren (85 %), sowie derjenigen, die im Homeoffice arbeiten (10 %).

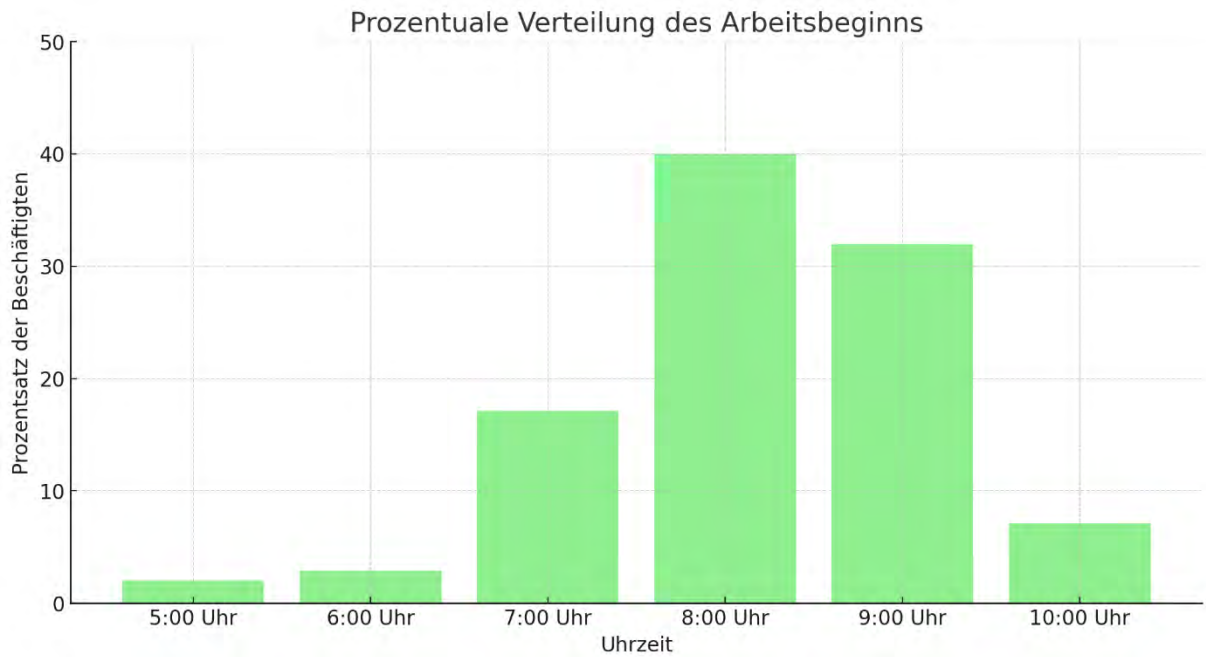


Abbildung 4 - Prozentuale Verteilung des Arbeitsbeginns der Frühschicht (Eigene Darstellung)

Die Verteilung der Beschäftigung rund um die Uhr (24/7) wurde durch eine Datenabstimmung zwischen den Daten des WSI (Wirtschafts- und Sozialwissenschaftliches Institut) und dem Industrielastprofil erreicht, wobei aufgrund der begrenzten Verfügbarkeit von Beschäftigungsdaten auf das Industrielastprofil zurückgegriffen wurde. Die Daten des WSI bieten lediglich Einblicke in den prozentualen Anteil der Beschäftigten, die an Samstagen, Sonn- und Feiertagen sowie in Schichtarbeit tätig sind. Angesichts der in Deutschland üblichen Fünf-Tage-Arbeitswoche wurde der prozentuale Anteil der Beschäftigung an Werktagen (Montag bis Freitag) durch das Industrielastprofil angenähert. Hierbei wurde die Annahme getroffen, dass das Verhältnis der Beschäftigten zum Energieverbrauch der industriellen Produktion korreliert.

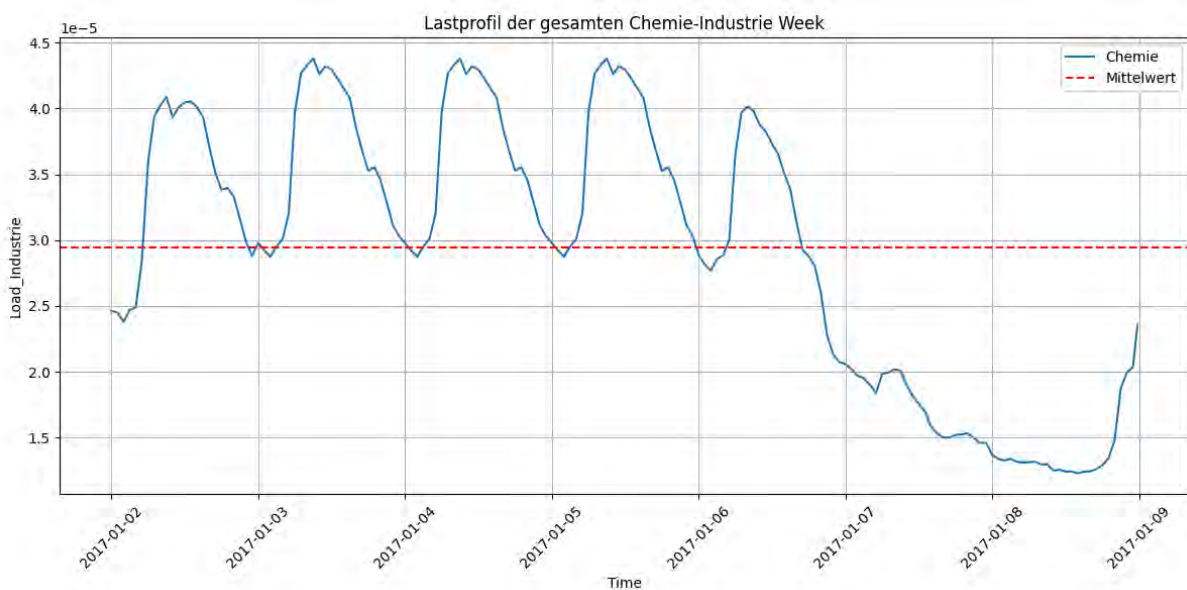


Abbildung 5 - genormtes Industrielastprofil über eine Woche (Eigene Darstellung)

Gemäß Abbildung 5 zeigt sich, dass die Tage mit erhöhtem Energieverbrauch insbesondere Dienstag bis Donnerstag sind. Aufgrund dieser Erkenntnis wurde für die Verteilung der Beschäftigten eine Gewichtung basierend auf dem Lastprofil vorgenommen. Aus dieser Analyse resultiert die prozentuale Verteilung der Beschäftigung auf die einzelnen Wochentage, wie in der dargestellten Abbildung 6 ersichtlich ist.

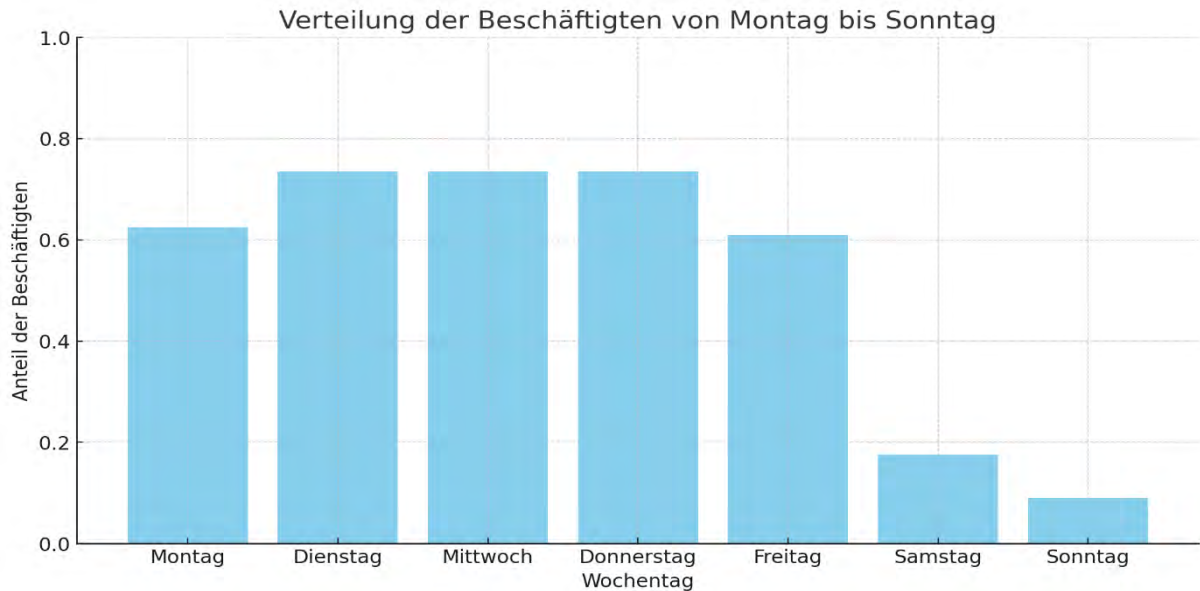


Abbildung 6 - Prozentuale Verteilung der Beschäftigten von Montag bis Sonntag (Eigene Darstellung)

Zusammenfassend ergibt sich aus den ermittelten Daten die folgende prozentuale Verteilung der Beschäftigung, welche in Abbildung 7 veranschaulicht wird. Beim Vergleich des Industrielastprofils aus Abbildung 5 mit der Beschäftigungsverteilung über eine Woche zeigt sich eine Korrelation zwischen den Werten. Dies deutet darauf hin, dass während der Hochlastzeiten (HLZ) ein hoher Anteil an verfügbaren BEV zur Verfügung steht.

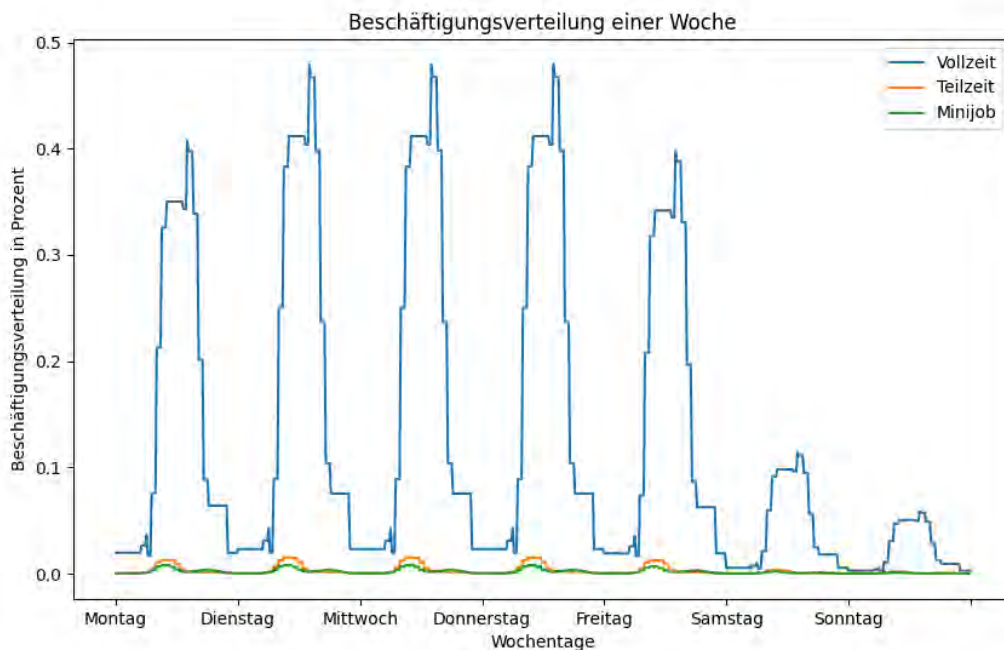


Abbildung 7 - Zusammengefasste Beschäftigungsverteilung über eine Woche (Eigene Darstellung)

Die prozentuale Verteilung der Vollzeitbeschäftigung über die gesamte Woche wurde in einer Excel-Tabelle angelegt. Dabei wird für jeden Arbeitsbeginn eine separate Excel-Tabelle generiert, die die Speicherverfügbarkeit kumuliert für den Arbeitszeitraum von 8 Stunden in 1-minütiger Auflösung darstellt. Diese Zeitreihen werden für die Python-Simulation in Kapitel 3.3 Entwicklung der Python-Ladesimulation eingelesen.

3.3 Entwicklung der Python-Ladesimulation

Die Simulation, die zur Untersuchung der Effekte des bidirektionalen Ladens von Elektrofahrzeugen auf die Spitzenlastreduktion durchgeführt wird, folgt einem detaillierten, mehrstufigen Prozess. Jeder Schritt ist darauf ausgerichtet, eine präzise Modellierung der realen Dynamiken und Potenziale zu gewährleisten. Im Folgenden werden die einzelnen Schritte des Simulationsprozesses erläutert.

3.3.1 Schritt 1 - Zeitliche Auflösung und Interpolation des Industrie-Lastprofils

Dieser Codeabschnitt in Abbildung 8 implementiert eine Prozedur zur Vorverarbeitung von Lastprofilen aus einer CSV-Datei (Comma-separated values), die für industrielle Anwendungen verwendet werden. Zunächst wird die CSV-Datei eingelesen und ein neues DataFrame erstellt, das einen kontinuierlichen Zeitbereich mit einer viertelstündigen Auflösung für das gesamte Jahr umfasst.

```

# Pfad zur CSV-Datei
csv_datei_pfad = r'../inputs/LP_weighted_normed_Industrie_ges_2017_hour.csv'

start_datum = '2017-01-01 00:00:00'
end_datum = '2017-12-31 23:59:00'
frequenz = '15min' # Neues DF hat Auflösung von 15 Minuten

# Auswahl I_1,I_4,I_5,I_6,I_7,I_8,I_9,I_10,I_11,I_12,I_13,I_0
ip_auswahl = ['I_0', 'I_1', 'I_5', 'I_6', 'I_13']

for ip in ip_auswahl:
    output_filename = f'../intermediates/LP_{ip}_{frequenz}_step_1.csv'
    print("Berechne:", output_filename, "...")

    # Quell-CSV-Datei einlesen
    df = pd.read_csv(csv_datei_pfad, usecols=[ip])

    # Erstellen eines Datumsbereichs von 2017-01-01 00:00:00 bis 2017-12-31 23:59:00 in 15 min Auflösung
    datumsbereich = pd.date_range(name='time', start=start_datum, end=end_datum, freq=frequenz)

    # Neues DataFrame erstellen, das auf dem Datumsbereich als Index basiert
    df_neu = pd.DataFrame(index=datumsbereich)

    # Annahme: Der ursprüngliche Index in 'df' ist in stündlicher Auflösung und beginnt ebenfalls am 2017-01-01 00:00:00
    df.index = pd.date_range(start=start_datum, periods=len(df), freq='h')

    # Die ursprünglichen Werte in das neue DataFrame einfügen (in die nächste Spalte neben dem Datum)
    df_neu = df_neu.join(df, how='left')

    # IP-Spalte umbenennen (soll später immer gleich heißen)
    df_neu.rename(columns={ip:'lp'}, inplace=True)

    # Lineare Interpolation der fehlenden Werte
    df_neu.ffill(inplace=True)

    sum_lp1 = df_neu.loc[:,].sum().values[0]
    if (not math.isclose(sum_lp1, 4.0)):
        df_neu['lp'] = (df_neu['lp'] / sum_lp1) * 4
        print(df_neu.loc[:,].sum())

    # Interpolierte und normierte Daten in eine neue CSV-Datei schreiben
    df_neu.to_csv(output_filename)
    print("    --> erfolgreich abgeschlossen")

```

Abbildung 8 – Schritt 1: Zeitliche Auflösung und Interpolation des Industrie-Lastprofils (Eigene Darstellung)

Dies ist erforderlich, um eine einheitliche Darstellung der Zeitreihe zu gewährleisten. Dann werden die originalen Lastprofilwerte aus der eingelesenen CSV-Datei in das neue DataFrame eingefügt. Da die ursprünglichen Daten nur eine stündliche Auflösung aufweisen, werden fehlende Werte durch lineare Interpolation, wie in Absatz 2.1.4 beschrieben, ersetzt, um eine kontinuierliche Zeitreihe zu erhalten, die für jede viertel Stunde des Jahres gültig ist. Schließlich werden die bearbeiteten Daten in eine neue CSV-Datei geschrieben, um die aktualisierte und vorverarbeitete Zeitreihe zu speichern, die nun eine 15-minütige Auflösung aufweist und für weitere Analysen oder Simulationen verwendet werden kann. Die 15-minütige Auflösung in der Simulation verschiedener Ladeszenarien für EVs mit bidirektionalem Laden ist von entscheidender Bedeutung, um präzise und effektive Ergebnisse zu erzielen. Dies ist insbesondere relevant, um die dynamischen Schwankungen im Stromnetz und die daraus resultierenden Anforderungen an das Lademanagement von Elektrofahrzeugen genau abzubilden.

3.3.2 Schritt 2: Berechne Tageshöchstwerte

Das Python-Skript "Berechne Tageshöchstwerte" in Abbildung 9 implementiert einen Algorithmus zur Analyse und Verarbeitung von normierten Lastprofilen industrieller Verbraucher. Ziel ist es, Tageshöchstwerte aus den zeitlich aufgelösten Daten zu extrahieren und weiterführende Analysen durchzuführen.

```

''' Version 3.0 Step_2 Berechne den Tageshöchstwert der CSV-Datei aus
Step_1 des jeweiligen normierten Lastprofils. Anwendung der Z-Score Normalisierung auf die Spalte prozentualer Anstieg
zur Normierung des prozentualen Anstiegs
Die Ergebnisse werden direkt in die CSV- Datei Step_2 geschrieben
Die Funktion glob() iteriert über alle Lastprofile die im Ordner intermediates LP_*__step_1.csv zu finden sind.
Auswahl eines bestimmten Lastprofils durch Ersetzen des * Zeichen in Zeile 50
Änderung zu Version 2: Berechne Tagesmittelwert
Änderung zu Version 1: iteriere über alle Lastprofile die du finden kannst '''

import pandas as pd
from scipy.stats import zscore

def calc(input_file):
    print("Berechne Tageshöchstwerte:", input_file)

    # CSV-Datei einlesen, Zeitstempel als Index
    df = pd.read_csv(input_file, parse_dates=[0], index_col=0)

    # Erstellen einer Kopie, um SettingWithCopyWarning zu vermeiden
    df_neu = df.copy()

    # Hinzufügen einer neuen Spalte für den täglichen Höchstwert im DataFrame
    df_neu['daily_max'] = float('nan') # Initialisieren mit NaN
    daily_max = df_neu.resample('D').max()

    # Aktualisieren der 'Täglicher Höchstwert'-Spalte mit dem berechneten täglichen Höchstwert
    for date, row in daily_max.iterrows():
        df_neu.loc[df_neu.index.date == date.date(), 'daily_max'] = row[df_neu.columns[0]]

    # Speichern der ausgewählten Spalten in einer neuen CSV-Datei
    out_filename = input_file.replace('__step_1.csv', '__step_2.csv')
    df_neu.to_csv(out_filename)

    print('--> Berechnung erfolgreich abgeschlossen:', out_filename)

```

Abbildung 9 - Schritt 2: Tageshöchstwerte (Eigene Darstellung)

Das Skript umfasst folgende Schritte:

- Datenrepräsentation und Vorverarbeitung: Die CSV-Datei mit den Lastprofilen wird eingelesen, wobei jeder Zeitpunkt als Index betrachtet wird. Dies ermöglicht eine zeitlich geordnete Darstellung der Verbrauchsdaten.
- Berechnung der Tageshöchstwerte: Das Skript berechnet für jeden Tag die maximale Lastspitze. Hierfür wird eine neue Spalte im DataFrame erstellt, um die täglichen Höchstwerte zu speichern.
- Ausgabe der Ergebnisse: Nach der Berechnung werden die Ergebnisse in einer neuen CSV-Datei gespeichert. Dies ermöglicht eine weitere Analyse und Visualisierung der Daten in anderen Anwendungen oder Umgebungen.
- Automatisierte Verarbeitung mehrerer Lastprofile: Das Skript wurde so konzipiert, dass es automatisch über alle verfügbaren Lastprofile iteriert und die oben genannten Analysen für jedes Profil durchführt. Dies ermöglicht eine effiziente Verarbeitung großer Datenmengen und erleichtert den Vergleich und die Untersuchung verschiedener Lastprofile.

Insgesamt stellt das Python-Skript eine wissenschaftliche Methode zur Analyse und Verarbeitung von Lastprofilen dar, die es ermöglicht, wichtige Kennzahlen und Muster im Energieverbrauchsverhalten industrieller Verbraucher zu identifizieren und zu verstehen.

3.3.3 Schritt 3: Berechnung der EV – Verfügbarkeiten

Die Methodik, die im vorgelegten Python-Skript in Abbildung 10 und Abbildung 11 verwendet wird, zielt darauf ab, die Verfügbarkeiten von Elektrofahrzeugen (EVs) über ein ganzes Jahr hinweg zu berechnen und zu modellieren, basierend auf der Verteilung der Ankunftszeiten der Fahrzeuge an

einem bestimmten Standort. Diese Berechnung wird genutzt, um das Potenzial von EV-Batterien als Zwischenspeicher im Stromnetz besser zu verstehen und zu quantifizieren.

```
def get_dow_name(datetime):
    match datetime.weekday():
        case 0: return 'Montag'
        case 1: return 'Dienstag'
        case 2: return 'Mittwoch'
        case 3: return 'Donnerstag'
        case 4: return 'Freitag'
        case 5: return 'Samstag'
        case 6: return 'Sonntag'

def calc_sheet(df, df_verteilung, verteilung_sheet_name, target_column_name):
    verteilung_sheet = df_verteilung.get(verteilung_sheet_name)

    # calculate battery status
    max_rows = len(df.index) - 1
    start_row_index = 0
    end_row_index = max_rows
    row_index = start_row_index
    while (row_index < end_row_index):
        print_calculation_status(row_index, start_row_index, end_row_index, f'berechne {verteilung_sheet_name}...')

        row_datetime = df['time'].iloc[row_index]
        row_time = row_datetime.time()
        dow_name = get_dow_name(row_datetime)
        verteilung_cell = verteilung_sheet.loc[verteilung_sheet['Zeit'] == row_time, dow_name]
        verteilung = verteilung_cell.values[0]

        # print('\n row_index:', row_index, \
        #       '\n row_datetime:', row_datetime, \
        #       '\n row_time:', row_time, \
        #       '\n dow_name:', dow_name, \
        #       '\n target_column_name:', target_column_name, \
        #       '\n verteilung:', verteilung)

        df.at[row_index, target_column_name] = verteilung.astype(float)
        row_index += 1
```

Abbildung 10 - Schritt 3: Berechnung der BEV – Verfügbarkeiten - Teil 1 (Eigene Darstellung)

Das Skript umfasst folgende Schritte:

- Datenmanagement und -vorbereitung: Import relevanter Bibliotheken: Zu Beginn werden pandas für die Datenmanipulation, datetime und timedelta für Zeitberechnungen sowie unterstützende Funktionen und Bibliotheken wie glob und threading importiert. glob wird genutzt, um mit Wildcards nach Dateipfaden zu suchen, während threading Parallelverarbeitung ermöglicht, um die Effizienz der Datenverarbeitung zu steigern.
- Einlesen der Beschäftigungsverteilung: Eine Excel-Datei, die die Beschäftigungsverteilung über verschiedene Tageszeiten und Wochentage enthält, wird eingelesen. Dies dient als Grundlage für die Bestimmung der EV-Verfügbarkeit basierend auf den Ankunftszeiten der Beschäftigten.
- get_dow_name diese Funktion übersetzt den Wochentag eines datetime-Objekts in den entsprechenden Namen auf Deutsch, was für die Zuordnung zu den Daten in der Beschäftigungsverteilung erforderlich ist.
- Hauptfunktion calc_sheet und calc_sheets: Die Funktion calc_sheet berechnet die Verfügbarkeiten für ein spezifisches Zeitfenster basierend auf der Beschäftigungsverteilung. calc_sheets nutzt diese Funktion, um die Verfügbarkeiten für mehrere definierte Zeitfenster zu berechnen und in der Ausgangsdatenstruktur zu speichern.


```
def calc_sheets(df_verteilung, df):
    calc_sheet(df, df_verteilung, '5.00 Uhr', '5:00')
    calc_sheet(df, df_verteilung, '6.00 Uhr', '6:00')
    calc_sheet(df, df_verteilung, '7.00 Uhr', '7:00')
    calc_sheet(df, df_verteilung, '8.00 Uhr', '8:00')
    calc_sheet(df, df_verteilung, '9.00 Uhr', '9:00')
    calc_sheet(df, df_verteilung, '10.00 Uhr', '10:00')
    calc_sheet(df, df_verteilung, '14.00 Uhr', '14:00')
    calc_sheet(df, df_verteilung, '22.00 Uhr', '22:00')
```

```
def calc(df_verteilung, in_filename):
    # Lastprofil aus Schritt 2 einlesen
    df = pd.read_csv(in_filename)

    # ensure that first column is 'date-time'
    df.iloc[:, 0] = pd.to_datetime(df.iloc[:, 0])

    # Verteilung der Verfügbarkeiten berechnen
    calc_sheets(df_verteilung, df)

    # neue Datei als Ergebnis von Schritt 3 abspeichern
    out_filename = in_filename.replace('__step_2.csv', '__step_3.csv')
    df.to_csv(out_filename, index = False)
    print(' --> Berechnung erfolgreich abgeschlossen:', out_filename)
```

```
import glob
import threading

# -----
excel_file_path = r'..\inputs\Beschäftigungsverteilung_1min.xlsx'
df_verteilung = pd.read_excel(excel_file_path, None)
# -----

threads = list()

# iteriere über alle Lastprofile die wir finden können...
for lp in glob.glob('..\intermediates\*_step_2.csv'):
    # calc(df_verteilung, lp)
    thread = threading.Thread(target=calc, args=(df_verteilung, lp))
    threads.append(thread)
    thread.start()

for index, thread in enumerate(threads):
    thread.join()
```

Abbildung 11 - Schritt 3: Berechnung der BEV – Verfügbarkeiten - Teil 2 (Eigene Darstellung)

- Anreicherung der Daten: Für jedes Zeitfenster werden zusätzliche Spalten in der Ausgangsdatenstruktur angelegt, welche die Verfügbarkeiten der EVs repräsentieren.
- Parallelisierung der Berechnung: Einsatz von threading, um die Rechenleistung zu optimieren, wird für jede zu verarbeitende CSV-Datei ein separater Thread gestartet. Dies ermöglicht es, die Berechnungen für mehrere Lastprofile parallel durchzuführen, was besonders bei der Verarbeitung großer Datenmengen von Vorteil ist.
- Finalisierung der Daten: Nach der Berechnung aller Verfügbarkeiten für die definierten Zeitfenster wird das Ergebnis in einer neuen CSV-Datei gespeichert, welche die erweiterten Daten für die weitere Analyse bereitstellt.

Durch die detaillierte Berücksichtigung von Beschäftigungsverteilungen und die zeitliche Anreicherung der Daten ermöglicht der Ansatz eine fundierte Analyse der Potenziale von Elektrofahrzeugen als flexible Energiespeicher. Die Integration von glob zur Dateiverwaltung und threading zur Effizienzsteigerung spiegelt zudem die Bedeutung von skalierbaren und leistungsstarken Datenverarbeitungstechniken in der Energieforschung wider.

3.3.4 Schritt 4: Berechnung der Batterie Zielzustände

Das Python-Skript in Abbildung 12 und Abbildung 13 ist darauf ausgerichtet, die Zielzustände der Batteriekapazität von EVs zu berechnen. Diese Berechnungen umfassen die maximale Kapazität der EV-Batterien, State of Charge (SoC) bei 80% (p80) bei der Ankunft der EVs und die minimale Batteriekapazität, die für das bidirektionale Laden nicht unterschritten werden darf.

```
def calc_battery_status_at_row(df, profile, year, time, row_index):
    max_status = df.at[row_index, time] * f_capacity(profile, year) # [GWh]
    max_in_1h = 0
    max_in_2h = 0
    p80_status = max_status * 0.8 # [GWh] der SoC den die Akkus bei der Ankunft an der Ladesäule haben
    min_status = max_status * 0.33 # [GWh] der minimale SoC der in den Akkus erhalten bleiben muss
    zeit_bis_max = 10 # 10*15 == 150 Minuten
    zeit_bis_p80 = 5 # 5*15 == 75 Minuten

    last_row_index = len(df.index) - 1
    row_index_A = min(row_index + zeit_bis_p80, last_row_index)
    row_index_B = min(row_index + zeit_bis_max, last_row_index)

    future_A = df.at[row_index_A, time]
    future_B = df.at[row_index_B, time]

    if (future_B == 0): # heißt: zum Zeitpunkt B stehen keine Fahrzeuge mehr auf dem Parkplatz
        min_status = p80_status

    if (future_A == 0): # heißt: schon zum Zeitpunkt a stehen keine Fahrzeuge mehr auf dem Parkplatz
        min_status = max_status

    df.at[row_index, f'battery_{time}_max'] = max_status
    df.at[row_index, f'battery_{time}_p80'] = p80_status
    df.at[row_index, f'battery_{time}_min'] = min_status
```

Abbildung 12 - Schritt 4: Berechnung der Batterie Zielzustände Teil 1 (Eigene Darstellung)

Die Methodik des Skripts lässt sich in mehrere Schlüsselschritte untergliedern:

- Berechnung der Batteriezustände bei der Ankunft: Die Funktion (`calc_battery_status_at_row`) berechnet auf Basis der Ankunftszeiten aus einem vorherigen Schritt (Step_3) die Zielzustände der Batterien für jede Zeile (repräsentativ für ein EV) in der Datenquelle. Diese Berechnungen beziehen sich auf spezifische Zeitpunkte und sind essenziell, um die Interaktion der EVs mit dem Stromnetz im Rahmen des bidirektionalen Ladens zu modellieren.
- Zeitliche Berechnungen: Für jedes Fahrzeug wird die Zeit bis zum Erreichen der maximalen und 80%igen Ladung (`zeit_bis_max` und `zeit_bis_p80`) bestimmt. Diese Zeitwerte sind wichtig, um festzulegen, bis zu welchem Zeitpunkt die Batterien entladen werden dürfen, ohne die Fähigkeit zu beeinträchtigen, bis zur Abfahrt der EVs vollständig geladen zu sein.
- Berücksichtigung von Abfahrtszeiten: Das Skript prüft mit Hilfe der berechneten Zeitpunkte Future A und Future B (entsprechend 80% und maximaler Ladung), ob Fahrzeuge zum geplanten Zeitpunkt noch am Standort sind. Dies ist entscheidend, um zu gewährleisten, dass die minimalen und maximalen Batteriekapazitäten den Anforderungen des bidirektionalen Ladens entsprechen.

```

def insert_columns(df: pd.DataFrame, time):
    df.insert(len(df.columns), f'battery_{time}_min', .0, False)
    df.insert(len(df.columns), f'battery_{time}_p80', .0, False)
    df.insert(len(df.columns), f'battery_{time}_max', .0, False)

def calc_battery_status(df: pd.DataFrame, profile, year, time):
    # neue Spalten anlegen...
    insert_columns(df, time)

    # calculate battery status
    max_rows = len(df.index) - 1
    start_row_index = 0
    end_row_index = max_rows
    row_index = start_row_index
    while (row_index < end_row_index):
        print_calculation_status(row_index, start_row_index, end_row_index, f'berechne {time} Uhr...')

        calc_battery_status_at_row(df, profile, year, time, row_index)
        row_index += 1

def calc(in_filename):
    profile = get_profile(in_filename)
    if (not profile in employed_industry):
        return

    for year in sorted(portion_ev.keys()):
        print('Berechne Akku-Zielzustände...', year, '@', profile)

        # Lastprofil aus Schritt 3 einlesen
        df = pd.read_csv(in_filename)

        # zielzustand berechnen
        calc_battery_status(df, profile, year, '5:00')
        calc_battery_status(df, profile, year, '6:00')
        calc_battery_status(df, profile, year, '7:00')
        calc_battery_status(df, profile, year, '8:00')
        calc_battery_status(df, profile, year, '9:00')
        calc_battery_status(df, profile, year, '10:00')
        calc_battery_status(df, profile, year, '14:00')
        calc_battery_status(df, profile, year, '22:00')

        # neue Datei als Ergebnis von Schritt 4 abspeichern
        '''Checkpoint - wähle das entsprechende Szenario'''
        # neue Datei als Ergebnis von Schritt 4 abspeichern
        out_filename = in_filename.replace('__step_3', f'__{year}__step_4')
        df.to_csv(out_filename, index = False)
        print(' --> Berechnung erfolgreich abgeschlossen:', out_filename)

```

Abbildung 13 – Schritt 4: Berechnung der Batterie Zielzustände Teil 2 (Eigene Darstellung)

- Die Funktion calc berechnet die Zielzustände von EV-Batterien zu bestimmten Zeitpunkten innerhalb eines Tages für verschiedene Jahre. Diese Zielzustände umfassen Informationen wie die maximale, minimale und die zu 80% geladene Kapazität der Batterien.
- Aktualisierung der Datensätze: Für jedes relevante Zeitfenster werden neue Spalten (für minimale, 80%ige und maximale Batteriekapazität) in den Datensatz eingefügt und mit den berechneten Werten aktualisiert.
- Speicherung der Ergebnisse: Nach Abschluss der Berechnungen für alle Zeitfenster und Fahrzeuge wird das aktualisierte Dataset in einer neuen CSV-Datei gespeichert. Diese Datei dient als Grundlage für weitere Analysen oder als Eingabe für nachgelagerte Schritte in der Untersuchung des Potenzials von bidirektionalem Laden.

Das Skript implementiert einen umfassenden Ansatz zur Bewertung der Rolle von EV-Batterien im Kontext des bidirektionalen Ladens, indem es detaillierte Berechnungen der Batteriezustände bei Ankunft und Abfahrt der Fahrzeuge durchführt. Durch die Anpassung der verfügbaren Kapazität und die Berücksichtigung der Lade- und Entladezeiten ermöglicht es eine präzise Modellierung der

Interaktion zwischen Elektrofahrzeugen und dem Stromnetz, was für die Planung und Optimierung von Energiesystemen von entscheidender Bedeutung ist.

3.3.5 Schritt 5: Berechnung der reellen Last und der täglichen Höchstlast

Das Python Skript in Abbildung 14 ist Teil eines sequenziellen Prozesses, in dem es das auf 4 normierte Lastprofil mit dem Energieverbrauch eines Jahres für jede Zeitreihe multipliziert und anschließend die tägliche Höchstlast für jeden Tag berechnet. Diese Daten werden in den darauffolgenden Schritten (6b bis 6d) verwendet, um die Algorithmen für Be- und Entladung von Elektrofahrzeugbatterien zu optimieren.

```
# Energiebedarf Industrie in GWh
energy_industry = {'I_0': {'2030': 220_000, \
                          '2035': 258_000, \
                          '2040': 318_000, \
                          '2045': 357_000}, \
                  'I_1': {'2030': 80_930, \
                          '2035': 91_930, \
                          '2040': 112_880, \
                          '2045': 126_720}, \
                  'I_5': {'2030': 10_900, \
                          '2035': 12_300, \
                          '2040': 15_200, \
                          '2045': 17_060}, \
                  'I_6': {'2030': 6_460, \
                          '2035': 7_340, \
                          '2040': 9_010, \
                          '2045': 10_110}, \
                  'I_13': {'2030': 42_940, \
                           '2035': 48_780, \
                           '2040': 59_892, \
                           '2045': 67_237} \
                  }

def calc_szenario(out_filename, df: pd.DataFrame, year, profile):
    df['lp'] *= energy_industry[profile][year]
    df['daily_max'] *= energy_industry[profile][year]

def handle_file(in_filename):
    profile = get_profile(in_filename)
    if (not profile in energy_industry):
        return

    for year in sorted(energy_industry[profile].keys()):
        if (in_filename.find(f'__{year}__step_4') == -1):
            continue

        df = pd.read_csv(in_filename)

        # Name der Ergebnis-Datei festlegen
        out_filename = in_filename.replace(f'__{year}__step_4', f'__{year}__step_5')

        # berechne
        calc_szenario(out_filename, df, year, profile)

        # Speichern der ausgewählten Spalten in einer neuen CSV-Datei
        df.to_csv(out_filename, index = False)
        print('Daily-Max-Berechnung', year, 'erfolgreich abgeschlossen - Result-File:', out_filename)
```

Abbildung 14 – Schritt 5: Berechnung der tatsächlichen Last und des täglichen Höchstwertes (Eigene Darstellung)

Das Skript durchläuft alle gefundenen Lastprofile (CSV-Dateien) für die Jahre 2030 bis 2045 und führt für jedes Industrieprofil eine Berechnung durch. Innerhalb der Funktion `calc_szenario`, wird das übergebene DataFrame (repräsentiert das Lastprofil) Zeile für Zeile durchlaufen. Für jede Zeile wird der Wert in der Spalte `lp` (Lastprofil) und `daily_max` mit dem `energy_industry`-Wert multipliziert, um das skalierte Lastprofil zu erhalten.

Nach Abschluss der Berechnung für ein Lastprofil, wird das modifizierte DataFrame in eine neue CSV-Datei geschrieben. Der Name der neuen Datei wird durch Ersetzen eines Teils des Originaldateinamens

generiert (z.B. von 15min__step_4_24_2030.csv zu 15min__step_5_24_7_2030.csv), um den Übergang zum nächsten Schritt zu kennzeichnen.

3.3.6 Schritt 6a: Algorithmus Simple Laden

Das Python-Skript in Abbildung 15 ist Teil 1 der Analyse, die sich mit der Bewertung der Auswirkungen des Einsatzes von Elektrofahrzeugen im industriellen Sektor befasst, insbesondere in Bezug auf die Ladevorgänge und deren Einfluss auf den Gesamtenergiebedarf. Das Ziel des Skripts ist es, ein Szenario zu simulieren, in dem Elektrofahrzeuge der in der Industrie beschäftigten Personen ohne Berücksichtigung von Netzkapazitäten oder Ladezeitfenstern aufgeladen werden. Hierbei wird der Schwerpunkt auf die Berechnung der Ladungsvorgänge und die daraus resultierende Gesamtlast zu verschiedenen Tageszeiten gelegt.

```
# Szenario A: Wir laden ohne rücksicht voll, sobald das Auto an die Ladesäule kommt...
# Ziel hier ist es die Fahrzeuge so schnell wie möglich voll zu laden.
def calc_charge_in_row(year, profile, df: pd.DataFrame, row_index: int, time):
    row = df.iloc[row_index]
    prev_soc_energy = get_previous_soc(df, row_index, time)
    max_soc_energy = row.at[f'battery_{time}_max']
    present_people_pct = row.at[time]
    max_charging_power = bev_charger_rate \
        * portion_ev[year] \
        * employed_industry[profile] \
        * present_people_pct \
        * efficiency

    charge_energy = 0
    if (prev_soc_energy < max_soc_energy):
        charge_energy = min(max_charging_power * aufloesung, max_soc_energy - prev_soc_energy)

    new_soc_energy = min(max_soc_energy, prev_soc_energy + charge_energy)

    df.at[row_index, f'battery_{time}_charge'] = charge_energy
    df.at[row_index, f'battery_{time}_soc'] = new_soc_energy

    return charge_energy

def calc_szenario(year, profile, df: pd.DataFrame):
    max_rows = len(df.index) - 1
    start_row_index = 0
    end_row_index = max_rows
    row_index = start_row_index
    while (row_index < end_row_index):
        print_calculation_status(row_index, start_row_index, end_row_index)

        lp_power = df.at[row_index, 'lp']
        total_energy = .0
        total_energy += calc_charge_in_row(year, profile, df, row_index, '5:00')
        total_energy += calc_charge_in_row(year, profile, df, row_index, '6:00')
        total_energy += calc_charge_in_row(year, profile, df, row_index, '7:00')
        total_energy += calc_charge_in_row(year, profile, df, row_index, '8:00')
        total_energy += calc_charge_in_row(year, profile, df, row_index, '9:00')
        total_energy += calc_charge_in_row(year, profile, df, row_index, '10:00')
        total_energy += calc_charge_in_row(year, profile, df, row_index, '14:00')
        total_energy += calc_charge_in_row(year, profile, df, row_index, '22:00')
        df.at[row_index, 'charge_energy'] = total_energy
        df.at[row_index, 'lp_total'] = lp_power + (total_energy / aufloesung)

        row_index += 1
```

Abbildung 15 – Schritt 6a: Algorithmus Simple Laden (Eigene Darstellung)

Variablen:

- employed_industry: Die Anzahl der in der Industrie angestellten Personen, welche für die Berechnung der potenziellen Anzahl an Elektrofahrzeugen herangezogen wird.
- portion_ev: Der Anteil der Elektrofahrzeuge im Straßenverkehr für ein bestimmtes Szenario. Dieser Wert wird verwendet, um die Gesamtanzahl der Elektrofahrzeuge abzuschätzen, die im Besitz der Industrieangestellten sind.

- `present_people_pct`: Anteil der Beschäftigten zum jeweiligen Zeitpunkt in der Zeitreihe.
- `aufloesung`: durch umwandlung von stündlich auf viertelstündlich.
- `max_charging_power`: maximale Ladeleistung (11KW/EV).
- `prev_soc_energy`: bevorzugte Ladestatus SoC der jeweiligen Zeit in der entsprechenden Zeitreihe
- `max_soc_energy`: maximale Kapazität der kumulierten Speicher.
- `charge_energy`: Ladeenergie die in die Speicher geladen werden.

Funktionen und Berechnungen:

- `calc_charge_in_row`: Berechnet die Ladung für ein spezifisches Zeitintervall (z.B. '5:00', '6:00' usw.) basierend auf dem aktuellen Ladestatus, dem maximal möglichen Ladestatus und dem maximalen Ladestrom, der durch den Anteil der EVs im Verkehr und die Anzahl der beschäftigten Personen in der Industrie bestimmt wird. Es wird sichergestellt, dass die Ladung nicht über den maximalen Status hinausgeht, und aktualisiert den Ladestatus sowie den Ladungswert im DataFrame.
- `calc_szenario`: Durchläuft jede Zeile des Data Frames (jede Zeile repräsentiert einen bestimmten Zeitpunkt) und summiert die Gesamtladung, die zu verschiedenen Zeitpunkten während des Tages erfolgt, zusammen mit dem ursprünglichen Lastprofil (`lp`), um ein Gesamtlastprofil (`lp_total`) zu erhalten.

Das Skript verfolgt einen iterativen Ansatz, bei dem für jedes Lastprofil (repräsentiert durch eine CSV-Datei) die zusätzliche Last durch das Laden von Elektrofahrzeugen zu verschiedenen Tageszeiten berechnet wird. Die Berechnung fokussiert ein "Worst-Case"-Szenario, in dem alle Fahrzeuge so schnell wie möglich geladen werden, ohne Rücksicht auf optimale Ladezeiten oder Netzkapazitäten. Die resultierende Gesamtlast wird dann verwendet, um die Auswirkungen dieses Ladeverhaltens auf die Energienachfrage zu analysieren.

Diese spezifische Analyse dient dazu, die potenziellen Spitzenlasten zu identifizieren, die entstehen könnten, wenn ein hoher Anteil der Elektrofahrzeuge in einem unkoordinierten oder suboptimalen Muster geladen wird. Die Erkenntnisse aus dieser Simulation können dazu beitragen, die Notwendigkeit von intelligenten Ladeinfrastrukturen und -strategien zu unterstreichen, um die Netzbelastung zu minimieren und eine effiziente Integration von Elektrofahrzeugen in das Energiesystem zu gewährleisten.

3.3.7 Schritt 6b: Algorithmus Laden mit Ladebremse

Das Python-Skript in Abbildung 16 beschreibt einen fortgeschrittenen Ladealgorithmus, genannt "Laden mit Ladebremse", für EVs im Kontext der Industriebeschäftigung und des voraussichtlichen Anteils von EVs im Straßenverkehr für ein bestimmtes Jahr. Die Kernidee dieses Szenarios ist es, die Fahrzeuge so schnell wie möglich vollständig zu laden, jedoch ohne die Gesamtlast über ein zuvor definiertes "normales" Tagesmaximum (Daily-Maximum) hinaus ansteigen zu lassen.

```

# Szenario B: Wie Variante A, jedoch mit 'Ladebremse':
# Wir laden sobald das Auto an die Ladesäule kommt, gehen bei der Last aber
# niemals über das 'normale' Daily-Maximum drüber. Auch hier sollen die
# Fahrzeuge so schnell wie möglich voll geladen werden.
def calc_charge_in_row(year, profile, df: pd.DataFrame, row_index: int, time, total_energy):
    row = df.iloc[row_index]

    present_people_pct = row.at[time]
    lp_power = row.at['lp']
    new_max_lp_power = row.at['daily_max'] * .99
    available_power = max(new_max_lp_power - lp_power, 0)
    max_charging_power = bev_charger_rate \
        * portion_ev[year] \
        * employed_industry[profile] \
        * present_people_pct \
        * efficiency

    max_soc_energy = row.at[f'battery_{time}_max']
    prev_soc_energy = get_previous_soc(df, row_index, time)
    charge_potential_energy = min(available_power, max_charging_power) * aufloesung
    charge_energy = .0
    if (prev_soc_energy + charge_potential_energy > max_soc_energy):
        charge_energy = max(max_soc_energy - prev_soc_energy, 0)
    else:
        charge_energy = charge_potential_energy

    total_charging_power = (total_energy - lp_power) / aufloesung
    target_charging_power = total_charging_power + (charge_energy / aufloesung)
    if ((lp_power + target_charging_power) > new_max_lp_power):
        charge_energy = max(0, new_max_lp_power - (lp_power + target_charging_power)) * aufloesung

    new_soc_energy = min(max_soc_energy, prev_soc_energy + charge_energy)

    df.at[row_index, f'battery_{time}_charge'] = charge_energy
    df.at[row_index, f'battery_{time}_soc'] = new_soc_energy

    return charge_energy

def calc_szenario(year, profile, df: pd.DataFrame):
    max_rows = len(df.index) - 1
    start_row_index = 0
    end_row_index = 4300# max_rows
    row_index = start_row_index
    while (row_index < end_row_index):
        print_calculation_status(row_index, start_row_index, end_row_index)

        lp_power = df.at[row_index, 'lp']
        total_energy = lp_power
        total_energy += calc_charge_in_row(year, profile, df, row_index, '5:00', total_energy)
        total_energy += calc_charge_in_row(year, profile, df, row_index, '6:00', total_energy)
        total_energy += calc_charge_in_row(year, profile, df, row_index, '7:00', total_energy)
        total_energy += calc_charge_in_row(year, profile, df, row_index, '8:00', total_energy)
        total_energy += calc_charge_in_row(year, profile, df, row_index, '9:00', total_energy)
        total_energy += calc_charge_in_row(year, profile, df, row_index, '10:00', total_energy)
        total_energy += calc_charge_in_row(year, profile, df, row_index, '14:00', total_energy)
        total_energy += calc_charge_in_row(year, profile, df, row_index, '22:00', total_energy)
        df.at[row_index, 'charge_energy'] = total_energy - lp_power
        df.at[row_index, 'lp_total'] = lp_power + (df.at[row_index, 'charge_energy'] / aufloesung)

        row_index += 1

```

Abbildung 16 -Schritt 6b: Algorithmus Laden mit Ladebremse (Eigene Darstellung)

Variablen:

- employed_industry: Anzahl der in der Industrie Beschäftigten.
- portion_ev: Anteil der Elektrofahrzeuge im Straßenverkehr für das Jahr 2045.
- present_people_pct: Anteil der Beschäftigten zum jeweiligen Zeitpunkt in der Zeitreihe.
- aufloesung: durch umwandlung von stündlich auf viertelstündlich.
- available_power: maximale verfügbare Ladeleistung die für jeden Zeitpunkt ermittelt wird.
- lp_power: Leistung aus dem Industrielastprofil der jeweiligen Zeitreihe.
- new_max_lp_power: Tägliche Höchstlast aus dem Industrielastprofil der jeweiligen Zeitreihe.
- max_charging_power: maximale Ladeleistung (11KW/EV).
- total_charging_power: Differenz der Spitzenlast daily_max und der Leistung lp_power begrenzt die maximale Ladeleistung max_charging_power.

- `prev_soc_energy`: bevorzugte Ladestatus SoC der jeweiligen Zeit in der entsprechenden Zeitreihe.
- `max_soc_energy`: maximale Kapazität der kumulierten Speicher.
- `charge_energy`: Ladeenergie die in die Speicher geladen werden.

Funktionen und Berechnungen:

- `calc_charge_in_row`: Nimmt das DataFrame (`df`), den Index der aktuellen Zeile (`row_index`), ein spezifisches Zeitintervall (`time`) und die Gesamtlast (`total`) zu Beginn des Zeitintervalls als Eingabe. Ermittelt den vorherigen Ladestatus und den maximalen Ladestatus der Batterie zu diesem Zeitpunkt, sowie das maximale Ladekapazität (`max_charge`), basierend auf dem Anteil der EVs und der Anzahl der Beschäftigten. Prüft, ob der vorherige Status unter dem maximalen Status liegt und ob die Gesamtlast unter dem täglichen Maximum ist. Wenn ja, wird geladen, wobei die Ladung das tägliche Maximum nicht überschreiten darf. Aktualisiert den Ladestatus und die Ladung im DataFrame und gibt die durchgeführte Ladung zurück.
- `calc_szenario`: Durchläuft alle Zeilen (repräsentieren Zeitintervalle) im DataFrame. Für jedes Zeitintervall berechnet die Funktion die zusätzliche Ladung durch Aufruf von `calc_charge_in_row` und summiert diese zum Gesamtlastprofil (`lp_total`).
- `handle_file`: Liest das Lastprofil aus einer CSV-Datei ein und bereitet es vor, indem es notwendige Spalten hinzufügt oder anpasst. Führt die Szenario-Berechnung durch und speichert die Ergebnisse in einer neuen CSV-Datei [`out_filename = save_result_filtered(df, in_filename, 'ladebremse')`].

Adaptives Laden: Durch die Berücksichtigung des täglichen Maximums wirkt der Algorithmus wie eine "Ladebremse", die verhindert, dass die Gesamtlast durch das Aufladen der EVs zu bestimmten Zeiten den Netzbedarf übermäßig erhöht.

Dynamische Anpassung: Der Algorithmus passt die Lademenge dynamisch an, basierend auf dem aktuellen Ladestatus der Batterien und der verfügbaren Kapazität im Stromnetz (repräsentiert durch das Daily-Maximum).

Optimierung: Ziel ist es, die Fahrzeuge so effizient wie möglich zu laden, so dass in Hochlastzeiten (HLZ) keine zusätzliche Spitzenlast erzeugt wird, was eine Balance zwischen der Notwendigkeit, Elektrofahrzeuge zu laden, und der Aufrechterhaltung der Netzstabilität schafft.

Der Algorithmus mit Ladebremse ist besonders relevant für Szenarien, in denen eine hohe Dichte von Elektrofahrzeugen erwartet wird und das Stromnetz auf die zusätzliche Last durch unkoordiniertes Laden vorbereitet sein muss. Durch die intelligente Steuerung des Ladevorgangs können Spitzenlasten vermieden und die Netzstabilität gewährleistet werden, was einen wichtigen Beitrag zur Integration von Elektromobilität in das Energiesystem darstellt.

3.3.8 Schritt 6c: Algorithmus Laden am Ende des Tages

Der "Laden End of Day"-Algorithmus in Abbildung 17 repräsentiert eine intelligente Ladestrategie für EVs in einem industriellen Kontext, bei der das Laden der Fahrzeuge gezielt auf das Ende des Arbeitstages gelegt wird, um die kumulierte Gesamtlast unter einem definierten Tagesmaximum zu halten. Diese Methode berücksichtigt sowohl die Notwendigkeit, die Fahrzeuge aufzuladen, als auch das Ziel, Spitzenlasten im Stromnetz zu vermeiden.

```
# Szenario C: Wir Laden erst am Ende des Arbeitstages. Dies kann dazu
# führen, dass einige Fahrzeuge nicht rechtzeitig voll geladen werden können.
def calc_charge_in_row(year, profile, df: pd.DataFrame, row_index: int, time, total_energy):
    row = df.iloc[row_index]
    new_max_lp_power = row.at['daily_max'] * 1.0
    prev_soc_energy = get_previous_soc(df, row_index, time)
    min_soc_energy = row.at[f'battery_{time}_min']
    max_soc_energy = row.at[f'battery_{time}_max']
    present_people_pct = row.at[time]
    max_charging_power = bev_charger_rate \
        * portion_ev[year] \
        * employed_industry[profile] \
        * present_people_pct \
        * efficiency

    charge_energy = 0
    if (prev_soc_energy < min_soc_energy):
        charge_energy = min(max_charging_power * aufloesung, min_soc_energy - prev_soc_energy)
        #print(charge_energy)

    new_soc_energy = min(max_soc_energy, prev_soc_energy + charge_energy)

    df.at[row_index, f'battery_{time}_charge'] = charge_energy
    df.at[row_index, f'battery_{time}_soc'] = new_soc_energy

    return charge_energy

def calc_szenario(year, profile, df: pd.DataFrame):
    max_rows = len(df.index) - 1
    start_row_index = 0
    end_row_index = 4300 #max_rows
    row_index = start_row_index
    while (row_index < end_row_index):
        print_calculation_status(row_index, start_row_index, end_row_index)

        lp_power = df.at[row_index, 'lp']
        total_energy = .0
        total_energy += calc_charge_in_row(year, profile, df, row_index, '5:00', total_energy)
        total_energy += calc_charge_in_row(year, profile, df, row_index, '6:00', total_energy)
        total_energy += calc_charge_in_row(year, profile, df, row_index, '7:00', total_energy)
        total_energy += calc_charge_in_row(year, profile, df, row_index, '8:00', total_energy)
        total_energy += calc_charge_in_row(year, profile, df, row_index, '9:00', total_energy)
        total_energy += calc_charge_in_row(year, profile, df, row_index, '10:00', total_energy)
        total_energy += calc_charge_in_row(year, profile, df, row_index, '14:00', total_energy)
        total_energy += calc_charge_in_row(year, profile, df, row_index, '22:00', total_energy)
        df.at[row_index, 'charge_energy'] = total_energy
        df.at[row_index, 'lp_total'] = lp_power + (total_energy / aufloesung)
        row_index += 1
```

Abbildung 17 – Schritt 6c: Algorithmus Laden am Ende des Tages (Eigene Darstellung)

Variablen:

- employed_industry: Anzahl der in der Industrie Beschäftigten.
- portion_ev: Anteil der Elektrofahrzeuge im Straßenverkehr für das Jahr 2045.
- present_people_pct: Anteil der Beschäftigten zum jeweiligen Zeitpunkt in der Zeitreihe.
- aufloesung: durch umwandlung von stündlich auf viertelstündlich.
- available_power: maximale verfügbare Ladeleistung die für jeden Zeitpunkt ermittelt wird.
- lp_power: Leistung aus dem Industrielastprofil der jeweiligen Zeitreihe.
- new_max_lp_power: Tägliche Höchstlast aus dem Industrielastprofil der jeweiligen Zeitreihe.
- max_charging_power: maximale Ladeleistung (11KW/EV).

- `total_charging_power`: Differenz der Spitzenlast `daily_max` und der Leistung `lp_power` begrenzt die maximale Ladeleistung `max_charging_power`.
- `prev_soc_energy`: bevorzugte Ladestatus SoC der jeweiligen Zeit in der entsprechenden Zeitreihe.
- `min_soc_energy`: minimale Kapazität der kumulierten Speicher.
- `max_soc_energy`: maximale Kapazität der kumulierten Speicher.
- `charge_energy`: Ladeenergie die in die Speicher geladen werden.

Funktionen und Berechnungen:

- `calc_charge_in_row`: Berechnet die Ladung für ein spezifisches Zeitintervall unter Berücksichtigung der bereits vorhandenen Gesamtlast und des Tagesmaximums. Der Algorithmus priorisiert das Laden der Fahrzeuge bis zum Mindestladestatus (`min_status`), solange die Gesamtlast das Tagesmaximum nicht überschreitet. Wenn der vorherige Ladestatus (`prev_status`) unter dem Mindestladestatus liegt und die Gesamtlast noch unter dem Tagesmaximum ist, wird geladen. Die Lademenge wird so angepasst, dass weder der Maximalstatus des Akkus überschritten wird noch die Gesamtlast das Tagesmaximum.
- `calc_szenario`: Durchläuft das Lastprofil (dargestellt als DataFrame) und summiert für jede Zeile (repräsentiert einen bestimmten Zeitpunkt) die zusätzliche Last durch das Laden der Fahrzeuge. Die Berechnung erfolgt zu mehreren spezifischen Zeitpunkten, um das End-of-Day-Laden zu simulieren. Die Gesamtladung zu jedem Zeitpunkt wird unter Berücksichtigung der vorherigen Ladung und des maximal zulässigen Tageslastmaximums angepasst.
- `handle_file`: Liest ein Lastprofil aus einer CSV-Datei, führt die Ladeszenario-Berechnungen durch und speichert die Ergebnisse in einer neuen Datei [`out_filename = save_result_filtered(df, in_filename, 'end-of-day')`]. Die Ausgabe informiert über den erfolgreichen Abschluss der Berechnungen.

Der "Laden End of Day"-Algorithmus zielt darauf ab, eine realistische und effiziente Ladestrategie für EVs in Industriegebieten zu modellieren, wobei die Notwendigkeit des Ladens der Fahrzeuge gegen die Anforderungen des Stromnetzes und die Vermeidung von Spitzenlasten abgewogen wird. Diese Strategie könnte besonders nützlich sein in Szenarien, in denen die Netzkapazität begrenzt ist und Spitzenlasten vermieden werden sollen. Die EVs werden nicht zwingend sofort nach Ankunft geladen.

Indem das Laden auf das Ende des Arbeitstages verschoben wird und dabei das Überschreiten eines definierten Tagesmaximums vermieden wird, trägt dieser Ansatz zur Netzstabilisierung bei und kann dazu beitragen, die Notwendigkeit für teure Netzausbauten zu reduzieren.

3.3.9 Schritt 6d: Algorithmus Bidirektionales Laden

Der Algorithmus für bidirektionales Laden ("Szenario BiDi") in Abbildung 18 ist eine fortschrittliche Methode zur Optimierung der Energielast in industriellen Umgebungen durch die Nutzung von Elektrofahrzeugbatterien nicht nur zum Laden, sondern auch zur Entladung in das Stromnetz. Dieser Ansatz zielt darauf ab, die Spitzenlasten zu reduzieren, indem temporäre Energie aus den

Fahrzeuggatterien ins Netz zuruckgespeist wird, wenn die Gesamtlast ein vordefiniertes Zielmaximum uberschreitet.

```

### Szenario D: Wie Variante 'Ladebremse', jedoch nutzen wir die Moglichkeit die Fahrzeuggatterien
# temporar zu entladen und so die Spitzenlasten zu drucken
def calc_charge_in_row(year, profile, df: pd.DataFrame, row_index: int, time, total_offset):
    row = df.iloc[row_index]

    present_people_pct = row.at[time]
    lp_power = row.at['lp']
    new_max_lp_power = row.at['daily_max'] * 0.995
    prev_charge_power = (total_offset - lp_power) / aufloesung
    available_power = new_max_lp_power - (prev_charge_power + lp_power)
    available_energy = available_power * aufloesung
    max_charging_power = bev_charger_rate \
        * portion_ev[year] \
        * employed_industry[profile] \
        * present_people_pct \
        * efficiency

    max_charging_energy = max_charging_power * aufloesung
    max_decharge_power = max_charging_power * 0.5
    max_decharge_energy = max_decharge_power * aufloesung * -1

    min_soc_energy = row.at[f'battery_{time}_min']

    change_energy = .0
    new_soc_energy = .0

    if (min_soc_energy > 0):
        max_soc_energy = row.at[f'battery_{time}_max']
        prev_soc_energy = get_previous_soc(df, row_index, time)
        charge_potential_energy = min(available_energy, max_charging_energy)
        if (prev_soc_energy > min_soc_energy):
            if (prev_soc_energy + charge_potential_energy > max_soc_energy):
                charge_energy = max(max_soc_energy - prev_soc_energy, 0)
            else:
                charge_energy = charge_potential_energy

            if (available_energy > 0):
                charge_energy = min(charge_energy, available_energy)
            else:
                charge_energy = max(max_decharge_energy, available_energy)
        #
        charge_energy = max(charge_energy, prev_soc_energy - min_soc_energy)

    new_soc_energy = min(max_soc_energy, prev_soc_energy + charge_energy)

    df.at[row_index, f'battery_{time}_charge'] = charge_energy
    df.at[row_index, f'battery_{time}_soc'] = new_soc_energy

    return charge_energy

```

Abbildung 18 – Schritt 6d: Algorithmus bidirektionales Laden (Eigene Darstellung)

Variablen:

- employed_industry: Anzahl der in der Industrie Beschäftigten.
- portion_ev: Anteil der Elektrofahrzeuge im Straßenverkehr für das Jahr 2045.
- present_people_pct: Anteil der Beschäftigten zum jeweiligen Zeitpunkt in der Zeitreihe.
- aufloesung: durch umwandlung von stundlich auf viertelstundlich.
- available_power: maximale verfügbare Ladeleistung die für jeden Zeitpunkt ermittelt wird.
- lp_power: Leistung aus dem Industrielastprofil der jeweiligen Zeitreihe.
- new_max_lp_power: Tägliche Höchstlast aus dem Industrielastprofil der jeweiligen Zeitreihe.
- max_charging_power: maximale Ladeleistung (11KW/EV).
- total_charging_power: Differenz der Spitzenlast daily_max und der Leistung lp_power begrenzt die maximale Ladeleistung max_charging_power.
- prev_soc_energy: bevorzugte Ladestatus SoC der jeweiligen Zeit in der entsprechenden Zeitreihe.
- min_soc_energy: minimale Kapazität der kumulierten Speicher.

- `max_soc_energy`: maximale Kapazität der kumulierten Speicher.
- `charge_energy`: Ladeenergie die in die Speicher geladen werden.

Funktionen und Berechnungen:

- `calc_charge_in_row`: Für jeden Zeitpunkt wird entschieden, ob geladen oder entladen wird, basierend auf der aktuellen Gesamtlast im Vergleich zum Zielmaximum. Das Laden erfolgt, wenn die aktuelle Last unter dem Zielmaximum liegt und der Ladestatus des Fahrzeugs unter dem maximalen Ladestatus. Die Lademenge wird so angepasst, dass das Zielmaximum nicht überschritten wird. Das Entladen (Rückspeisen ins Netz) wird aktiviert, wenn die Gesamtlast das Zielmaximum übersteigt und der Ladestatus über dem Mindestladestatus liegt. Die Menge der Rückspeisung wird so angepasst, dass die maximale Kapazität erreicht, aber nicht unterschritten wird.
- `calc_szenario`: Durchläuft alle Zeitpunkte des Lastprofils und berechnet für jeden die zusätzliche Last oder Rückspeisung durch das Laden bzw. Entladen der Elektrofahrzeuge. Das Ziel ist es, die Gesamtlast so nah wie möglich am definierten Zielmaximum zu halten, um Spitzenlasten zu vermeiden.
- `handle_file`: Liest das Lastprofil aus einer CSV-Datei, wendet den bidirektionalen Ladealgorithmus an und speichert die Ergebnisse in einer neuen Datei [`out_filename = save_result_filtered(df, in_filename, 'bidi')`]. Dabei wird für jedes Lastprofil die optimierte Gesamtlast dokumentiert.

Der bidirektionale Ladealgorithmus bietet eine dynamische Lösung zur Verwaltung der Energiebelastung in industriellen Umgebungen, indem er Elektrofahrzeuge als temporäre Energiespeicher nutzt. Dieser Ansatz kann dazu beitragen, die Abhängigkeit von traditionellen Spitzenlastkraftwerken zu verringern und die Nutzung erneuerbarer Energien zu optimieren, indem überschüssige Energie zeitweise in den Fahrzeugbatterien gespeichert und bei Bedarf wieder abgerufen wird. Insbesondere in Zeiten hoher Energienachfrage oder bei fluktuierender Energieerzeugung durch erneuerbare Quellen bietet diese Technologie eine effiziente Möglichkeit, die Netzstabilität zu unterstützen und die Energiekosten zu senken.

3.3.10 Schritt 7: Erstellung der Diagramme

Das Python-Skript in Abbildung 19 und Abbildung 20 ist darauf ausgelegt, die Ergebnisse einer Energiesimulation für einen bestimmten Zeitraum zu visualisieren. Die Simulationsergebnisse beziehen sich auf die Energielast und den Batteriezustand SoC (Ladung/Entladung) in einem industriellen Umfeld, das Elektrofahrzeuge einbezieht. Der Schwerpunkt liegt auf der Darstellung von Veränderungen im Tagesverlauf und dem Vergleich des Original-Lastprofils mit dem modifizierten Lastprofil unter Berücksichtigung des Ladens von Elektrofahrzeugen.

- Konvertierung von Zeitangaben: Start- und Enddatum werden in `DateTime`-Objekte umgewandelt, um den Zeitrahmen für die Analyse festzulegen.
- `prepare_results(df)`: Anpassung der Auflösung von Energie in Leistung

- Einlesen der Daten: Die Daten werden aus CSV-Dateien geladen, wobei angenommen wird, dass die erste Spalte Zeitstempel enthält, die in DateTime-Objekte konvertiert werden.

```
# Konvertieren der Start- und Enddaten in DateTime-Objekte
start_date = pd.to_datetime('2017-02-02 00:00:00')
end_date = pd.to_datetime('2017-02-02 23:59:00')

def prepare_results(df):
    df['battery_5:00_charge'] /= aufloesung
    df['battery_6:00_charge'] /= aufloesung
    df['battery_7:00_charge'] /= aufloesung
    df['battery_8:00_charge'] /= aufloesung
    df['battery_9:00_charge'] /= aufloesung
    df['battery_10:00_charge'] /= aufloesung
    df['battery_14:00_charge'] /= aufloesung
    df['battery_22:00_charge'] /= aufloesung

def plot(df, column_offset, y_label, title):
    # Filtern der Daten für den gewünschten Zeitraum
    filtered_data = df[(df.iloc[:, 0] >= start_date) & (df.iloc[:, 0] <= end_date)]

    plt.figure(figsize=(12, 6))
    plt.plot(filtered_data.iloc[:, 0], filtered_data.iloc[:, column_offset], label="5:00")
    plt.plot(filtered_data.iloc[:, 0], filtered_data.iloc[:, column_offset + 2], label="6:00")
    plt.plot(filtered_data.iloc[:, 0], filtered_data.iloc[:, column_offset + 4], label="7:00")
    plt.plot(filtered_data.iloc[:, 0], filtered_data.iloc[:, column_offset + 6], label="8:00")
    plt.plot(filtered_data.iloc[:, 0], filtered_data.iloc[:, column_offset + 8], label="9:00")
    plt.plot(filtered_data.iloc[:, 0], filtered_data.iloc[:, column_offset + 10], label="10:00")
    plt.plot(filtered_data.iloc[:, 0], filtered_data.iloc[:, column_offset + 12], label="14:00")
    plt.plot(filtered_data.iloc[:, 0], filtered_data.iloc[:, column_offset + 14], label="22:00")

    plt.legend()
    plt.xlabel('Time')
    plt.ylabel(y_label)
    plt.title(title)
    plt.grid(True)
    plt.xticks(rotation=45) # Dreht die X-Achsen-Beschriftungen, falls notwendig
    plt.tight_layout() # Verbessert die Layout-Anpassung
    plt.show()
```

Abbildung 19 – Schritt 7 Teil 1: Erstellung der Diagramme (Eigene Darstellung)

```

def plot_total(df, title):
    # Filtern der Daten für den gewünschten Zeitraum
    filtered_data = df[(df.iloc[:, 0] >= start_date) & (df.iloc[:, 0] <= end_date)]

    plt.figure(figsize=(12, 6))
    plt.plot(filtered_data.iloc[:, 0], filtered_data.loc[:, 'lp'], label="LP Original")
    plt.plot(filtered_data.iloc[:, 0], filtered_data.loc[:, 'lp_total'], label="LP Gesamt")

    plt.legend()
    plt.xlabel('Time')
    plt.ylabel('Leistung [GW]')
    plt.title(title)
    plt.grid(True)
    plt.xticks(rotation=45) # Dreht die X-Achsen-Beschriftungen, falls notwendig
    plt.tight_layout() # Verbessert die Layout-Anpassung
    plt.show()

def plot_battery():
    df = pd.read_csv('..\intermediates\LP_I_0_1min_step_4_24_7_2040.csv')
    # ensure that first column is 'date-time'
    df.iloc[:, 0] = pd.to_datetime(df.iloc[:, 0])

    # Filtern der Daten für den gewünschten Zeitraum
    filtered_data = df[(df.iloc[:, 0] >= start_date) & (df.iloc[:, 0] <= end_date)]

    plt.figure(figsize=(12, 4))
    plt.plot(filtered_data.iloc[:, 0], filtered_data.loc[:, 'battery_5:00_min'], label="battery_5:00_min")
    plt.plot(filtered_data.iloc[:, 0], filtered_data.loc[:, 'battery_6:00_min'], label="battery_6:00_min")
    plt.plot(filtered_data.iloc[:, 0], filtered_data.loc[:, 'battery_7:00_min'], label="battery_7:00_min")
    plt.plot(filtered_data.iloc[:, 0], filtered_data.loc[:, 'battery_8:00_min'], label="battery_8:00_min")
    plt.plot(filtered_data.iloc[:, 0], filtered_data.loc[:, 'battery_9:00_min'], label="battery_9:00_min")
    plt.plot(filtered_data.iloc[:, 0], filtered_data.loc[:, 'battery_10:00_min'], label="battery_10:00_min")
    plt.plot(filtered_data.iloc[:, 0], filtered_data.loc[:, 'battery_14:00_min'], label="battery_14:00_min")
    plt.plot(filtered_data.iloc[:, 0], filtered_data.loc[:, 'battery_22:00_min'], label="battery_22:00_min")

    plt.legend()
    plt.xlabel('Time')
    plt.ylabel('Speicherstatus SoC [GWh]')
    plt.title('Batterie-Zielzustand')
    plt.grid(True)
    plt.xticks(rotation=45) # Dreht die X-Achsen-Beschriftungen, falls notwendig
    plt.tight_layout() # Verbessert die Layout-Anpassung
    plt.show()

```

Abbildung 20 - Schritt 7 Teil 2: Erstellung der Diagramme (Eigene Darstellung)

Das Skript definiert mehrere Funktionen zur Visualisierung der Daten:

- plot: Visualisiert die Simulationsergebnisse für spezifische Zeitintervalle innerhalb des definierten Tages. Die Funktion plottet die Ergebnisse der unterschiedlichen Ladeszenarien, wieviel Energie beladen oder entladen wird. Es werden mehrere Zeitintervalle, wie in Abbildung 21 dargestellt, abgebildet, um den Verlauf und Vergleich zu erleichtern.

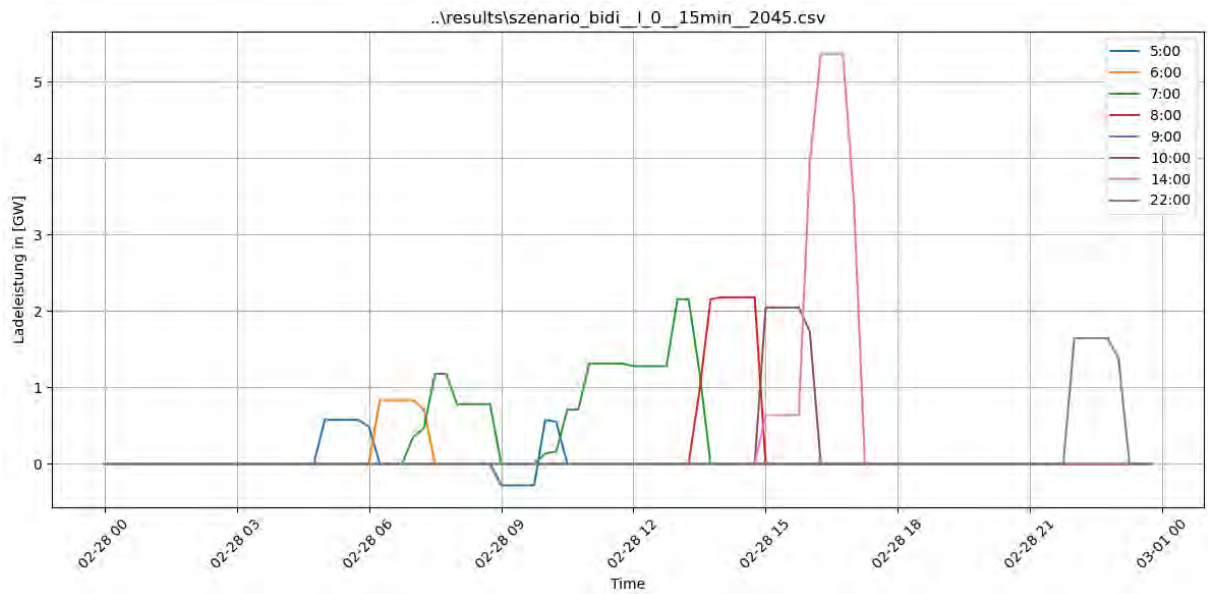


Abbildung 21 – Simulationsergebnisse des bidirektionalen Ladens in GW (Eigene Darstellung)

- plot_battery: Abbildung 22 zeigt den Zustand der Batteriekapazität zu verschiedenen Zeitpunkten des Tages. Diese Visualisierung ist besonders nützlich, um zu beurteilen, wie die Batterieladezustände sich über den Tag verteilen und inwiefern sie die festgelegten Mindestanforderungen erfüllen.

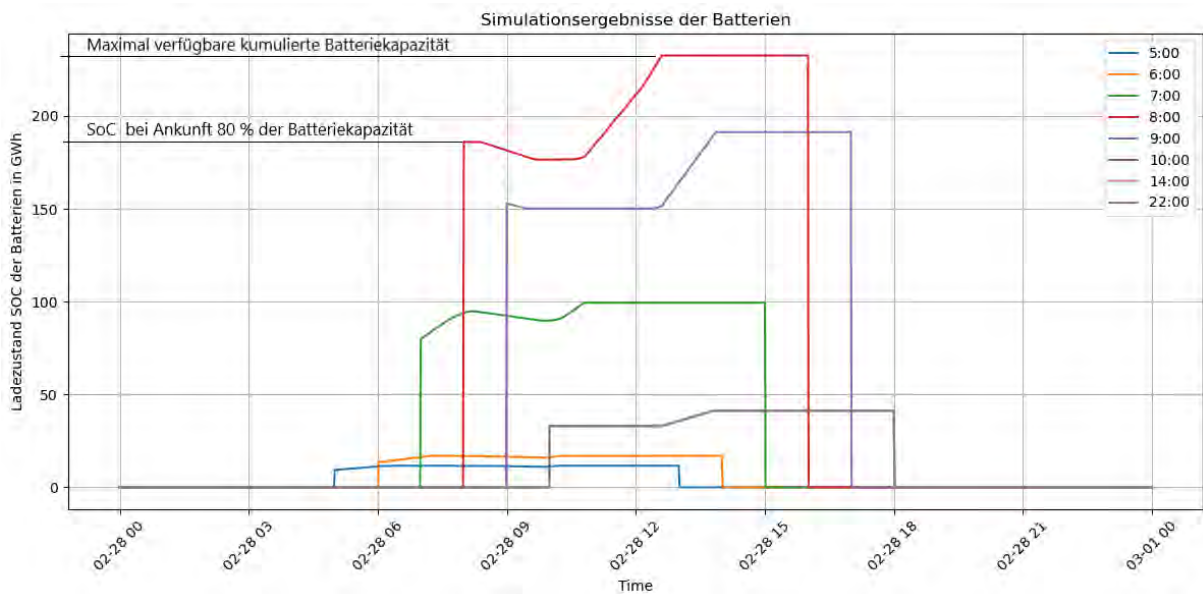


Abbildung 22 – Simulationsergebnis Ladezustand SoC bidirektionales Laden in GWh (Eigene Darstellung)

- plot_total: Abbildung 23 vergleicht das ursprüngliche Lastprofil mit dem modifizierten Lastprofil, das das Laden der Elektrofahrzeuge berücksichtigt. Ziel ist es, die Auswirkungen des Ladens auf den Gesamtenergieverbrauch im Tagesverlauf zu visualisieren.

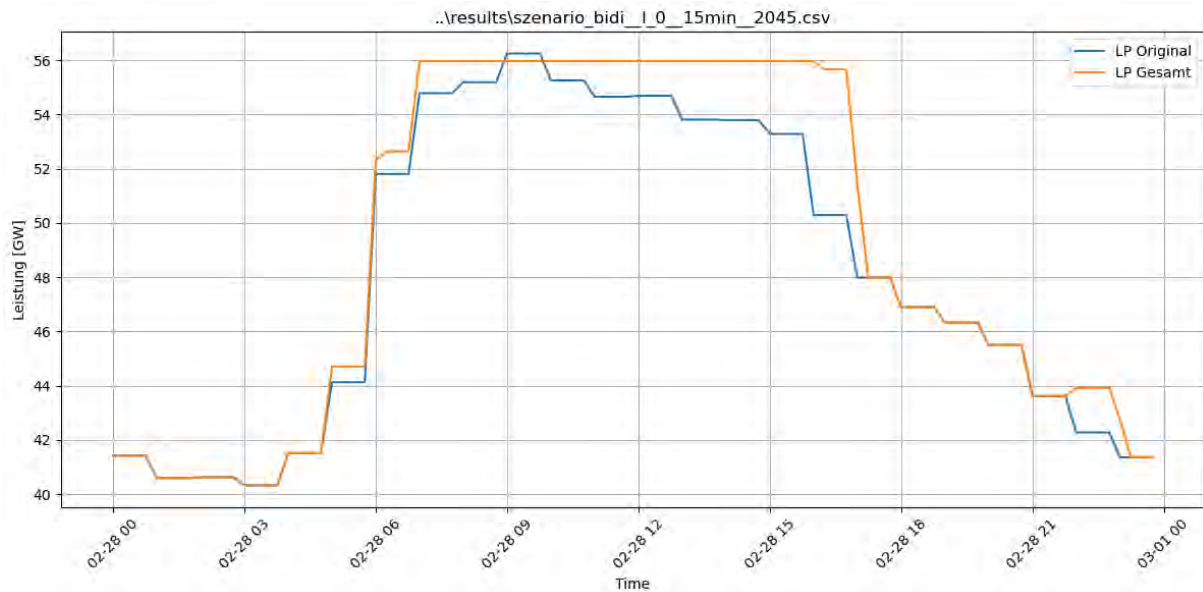


Abbildung 23 – Vergleich original Lastprofil und Lastprofil mit bidirektionalem Laden in GW (Eigene Darstellung)

Visualisierungsprozess: Für jede gefundene Datei im spezifizierten Verzeichnis werden die definierten Visualisierungsfunktionen aufgerufen, um die verschiedenen Szenarien der Energielast und des Batteriemangements darzustellen.

Analyse und Bewertung: Das Skript ermöglicht eine detaillierte Analyse und Bewertung der Auswirkungen von Elektrofahrzeugladeprozessen auf die Energielast in einem industriellen Kontext. Es hilft bei der Identifizierung von Spitzenlasten, der Bewertung der Effektivität von Ladestrategien und der Optimierung des Energiemanagements.

Anschauliche Darstellung: Durch die visuelle Aufbereitung der Simulationsergebnisse können komplexe Datenmuster leichter erkannt und interpretiert werden. Dies unterstützt Entscheidungsträger, Ingenieure und Forscher dabei, fundierte Strategien für das Energiemanagement und die Integration von Elektrofahrzeugen in industriellen Umgebungen zu entwickeln.

3.3.11 Szenario Helpers Skript

Das Skript in Abbildung 24 ist ein Hilfsprogramm für die Verwaltung und Analyse von Daten im Zusammenhang mit der Ladung von EVs und deren Batteriestatus über einen bestimmten Zeitraum. Es verwendet die Pandas-Bibliothek für die Datenmanipulation und -analyse in Python. Nachfolgend eine Übersetzung und Erläuterung der Kernfunktionalitäten des Skripts:


```

import pandas as pd
import math

# -----

bev_energy_gwh = 6e-5 # Batteriekapazität EV Mittelklasse in GWh Mittelklasse == 60 kWh
bev_charger_rate = 1.1e-5 # 3-phase EV charger with 11 kW
# Angestellte in der Industrie https://www.destatis.de/DE/Themen/Arbeit/Arbeitsmarkt/Erwerbstaetigkeit/Tabellen/eckwerttabelle.html
employed_industry = {'I_0': 8_000_000, \
                    'I_1': 228_000, \
                    'I_5': 786_000, \
                    'I_6': 956_000, \
                    'I_13': 466_500, \
                    }

# Anteil der Elektrofahrzeuge im Strassen Verkehr je Betrachtungsjahr
portion_ev = {'2030': .3, \
             '2035': .68, \
             '2040': .78, \
             '2045': .9}

efficiency = 0.9
aufloesung = 1 / 4 # 15 Minuten Scheiben

# -----

def f_capacity(profile, year):
    return bev_energy_gwh * employed_industry[profile] * portion_ev[year]

# -----

# insert columns per arrival time...
def insert_columns(df: pd.DataFrame, time):
    df.insert(len(df.columns), f'battery_{time}_charge', .0, False)
    df.insert(len(df.columns), f'battery_{time}_soc', .0, False)

# -----

# insert all szenario columns...
def prepare_columns(df: pd.DataFrame):
    df.insert(len(df.columns), 'charge_energy', .0, False)
    df.insert(len(df.columns), 'lp_total', .0, False)
    insert_columns(df, '5:00')
    insert_columns(df, '6:00')
    insert_columns(df, '7:00')
    insert_columns(df, '8:00')
    insert_columns(df, '9:00')
    insert_columns(df, '10:00')
    insert_columns(df, '14:00')
    insert_columns(df, '22:00')

# -----

```

Abbildung 24 - Szenario Helpers Teil 1 (Eigene Darstellung)

Variablen:

- bev_energy_gwh: Die Batteriekapazität eines mittelklassigen Elektrofahrzeugs in Gigawattstunden (GWh).
- bev_charger_rate_gwh: Die Laderate eines 3-phasigen EV-Ladegeräts mit 11 kW.
- efficiency: Der Wirkungsgrad des Ladeprozesses.
- aufloesung: Die Auflösung der Zeit, hier minütlich, für die Berechnung der maximalen Lademenge pro Zeiteinheit.

Hilfsfunktionen:

- get_bev_energy(): Gibt die gemittelte Batteriekapazität der EVs zurück.
- get_f_capacity(): Eine alternative Funktion, die die Batteriekapazität der EVs zurückgibt.

Spaltenverwaltung in DataFrames:

- insert_columns(df, time): Fügt dem DataFrame df zwei neue Spalten hinzu, eine für die Ladung und eine für den Status der Batterie zu einem bestimmten Zeitpunkt (time).

- `prepare_columns(df)`: Bereitet das DataFrame `df` vor, indem es Spalten für den Gesamtlastpunkt (`lp_total`) und für verschiedene Zeitpunkte durch Aufruf von `insert_columns()` hinzufügt.

Status der vorherigen Ladung:

- `get_previous_status(df, row_index, time)`: Ermittelt den vorherigen Status der Batterie zu einem gegebenen Zeitpunkt `time` basierend auf dem aktuellen und vorherigen Zeilenindex im DataFrame `df`. (Abbildung 25)

```
# -----
def get_previous_soc(df: pd.DataFrame, row_index, time):
    if (row_index == 0):
        return 0

    previous = df.iloc[row_index - 1][f'battery_{time}_soc']
    if (math.isclose(previous, .0)):
        return df.iloc[row_index][f'battery_{time}_p80']

    return previous

# -----

def print_results(df: pd.DataFrame, intermediate_filename, out_filename, szenario_name):
    sum_lp = df.loc[:, 'lp'].sum()
    sum_charges = df.loc[df['charge_energy'] > 0, 'charge_energy'].sum() #321.56843708160125
    #sum_charges = df.loc[:, 'charge_energy'].sum()
    sum_totals = df.loc[:, 'lp_total'].sum()
    print('\n-----', \
          '\nBerechnung Szenario "', szenario_name, '" erfolgreich abgeschlossen!', \
          '\n intermediate file: ', intermediate_filename, \
          '\n output file: ', out_filename, \
          '\n sum lp: {0:.1f}'.format(sum_lp * aufloesung), \
          '\n sum charges: {0:.1f}'.format(sum_charges), \
          '\n sum lp_totals: {0:.1f}'.format(sum_totals * aufloesung), \
          '\n-----\n')
```

Abbildung 25 - Szenario Helpers Teil 2 (Eigene Darstellung)

Ergebnisse speichern:

- `save_result_filtered(df, intermediate_filename, szenario_name)`: Speichert die gefilterten Ergebnisse in einer neuen Datei. Die Funktion passt den Dateinamen an das gegebene Szenario an und behält nur relevante Spalten bei.

Berechnungsstatus:

- `print_calculation_status(row_index, start_row_index, end_row_index)`: Gibt den Fortschritt der Berechnung aus, indem der aktuelle Stand basierend auf dem aktuellen Zeilenindex im Verhältnis zum Gesamtumfang der Daten angezeigt wird.
- Entfernen irrelevanter Ergebnis-Spalten und Speicher der Ergebnisse in einer neuen Datei.

```

def save_result_filtered(df: pd.DataFrame, intermediate_filename, szenario_name):
    # Name der Ergebnis-Datei festlegen...
    out_filename = intermediate_filename \
        .replace('\\intermediates\\LP__', f'\\results\\szenario_{szenario_name}__') \
        .replace('/intermediates/LP__', f'/results/szenario_{szenario_name}__') \
        .replace('__step_5', '')

    # Print some basic information
    print_results(df, intermediate_filename, out_filename, szenario_name)

    df_neu = df.filter(['time', \
                        'lp', \
                        'charge_energy', \
                        'lp_total', \
                        'battery_5:00_charge', \
                        'battery_5:00_soc', \
                        'battery_6:00_charge', \
                        'battery_6:00_soc', \
                        'battery_7:00_charge', \
                        'battery_7:00_soc', \
                        'battery_8:00_charge', \
                        'battery_8:00_soc', \
                        'battery_9:00_charge', \
                        'battery_9:00_soc', \
                        'battery_10:00_charge', \
                        'battery_10:00_soc', \
                        'battery_14:00_charge', \
                        'battery_14:00_soc', \
                        'battery_22:00_charge', \
                        'battery_22:00_soc'])

    print("   Schreibe Ergebnis-File:", out_filename)
    df_neu.to_csv(out_filename, index = False)

    return out_filename

# -----

def print_calculation_status(row_index, start_row_index, end_row_index, message = 'Berechnung läuft... '):
    # alle 10.000 Zeilen schreiben wir einen Log-Eintrag...
    if ((row_index % 10_000) == 0):
        a = row_index - start_row_index
        b = end_row_index - start_row_index
        print(message, '{0:.1f}'.format((a / b) * 100), '% -- aktuelle Zeile:', row_index)

# -----

def get_profile(filename):
    profile_start_index = filename.find('LP_I_') + 4
    profile_end_index = filename.find('__', profile_start_index)
    return filename[profile_start_index:profile_end_index]

```

Abbildung 26 - Szenario Helpers Teil 3 (Eigene Darstellung)

Ergebnisse speichern:

- `save_result_filtered(df, intermediate_filename, szenario_name)`: Speichert die gefilterten Ergebnisse in einer neuen Datei. Die Funktion passt den Dateinamen an das gegebene Szenario an und behält nur relevante Spalten bei.

Berechnungsstatus:

- `print_calculation_status(row_index, start_row_index, end_row_index)`: Gibt den Fortschritt der Berechnung aus, indem der aktuelle Stand basierend auf dem aktuellen Zeilenindex im Verhältnis zum Gesamtumfang der Daten angezeigt wird.

Verwaltung der Industrieprofile:

- `get_profile(filename)`: verwaltet die Industrieprofile in den CSV-Dateien bei der Verarbeitung der jeweiligen Schritte.

Dieses Skript ist für die Verwaltung von Funktionen, die in mehreren Python Skripten verwendet werden und wird über die Import Funktion (`from szenario_helpers import *`) im jeweiligen Skript aufgerufen.

4 Auswertung der Simulationsergebnisse

In diesem Kapitel erfolgt eine detaillierte Analyse der Simulationsergebnisse verschiedener Szenarien. Zunächst wird das Potenzial der Gesamtindustrie evaluiert. Anschließend konzentriert sich die Untersuchung auf die Diversität der Industriebranchen, um das Verhältnis zwischen potenzieller Energiespeicherkapazität und dem tatsächlichen Energiekonsum zu erforschen.

4.1 Simulation der gesamten Industrie Deutschlands

Im Jahr 2023 waren in der deutschen Industrie etwa 8 Millionen Menschen beschäftigt, verteilt auf circa 24.537 Unternehmen. [26] Der Gesamtenergieverbrauch an Strom dieses Sektors belief sich im Jahr 2020 auf annähernd 223 TWh/a. Angesichts des fortschreitenden Trends zur Elektrifizierung innerhalb der industriellen Prozesse und der zunehmenden Implementierung von Technologien zur Effizienzsteigerung wird prognostiziert, dass der Energiebedarf der deutschen Industrie bis zum Jahr 2045 auf bis zu 357 TWh/a ansteigen wird, wie in der folgenden Abbildung 27 zu sehen ist. [20]

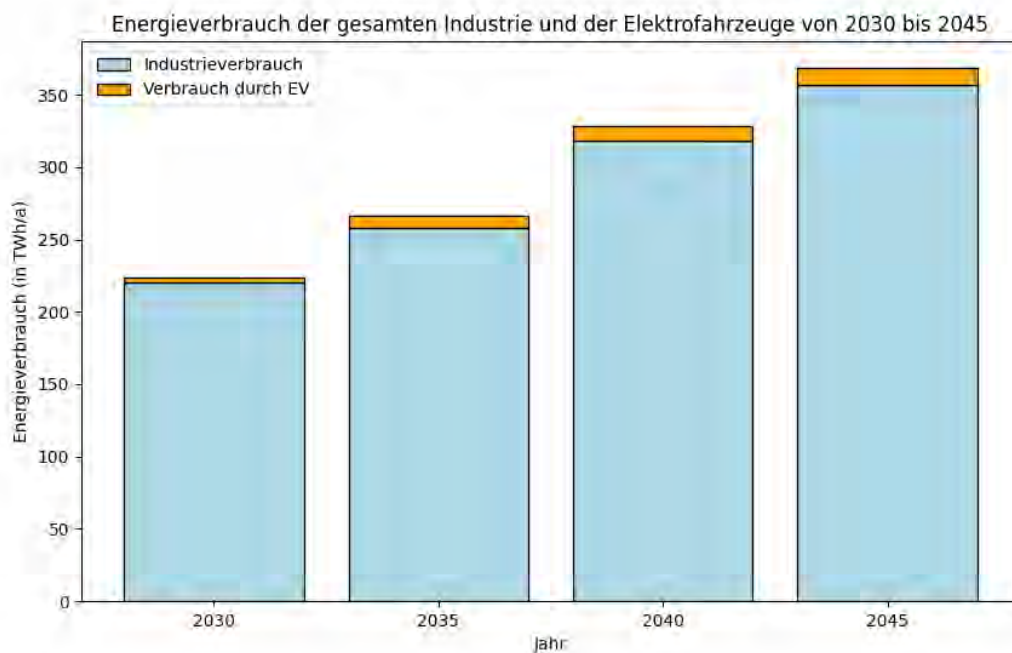


Abbildung 27 – Energieverbrauch Strom in TWh/a der gesamten Industrie von 2030 bis 2045 (Eigene Darstellung) [20]

Unter Zugrundelegung der dargelegten Daten zum gesamten Industriesektor und in Anbetracht der Entwicklungen im Bereich der Elektromobilität, wie in Abschnitt 3.1 erörtert, wurde eine Simulationsstudie für den Zeitraum von 2030 bis 2045 in Fünfjahresintervallen durchgeführt. In sämtlichen betrachteten Szenarien wurde mit dem Beschäftigungsprofil 24/7 gerechnet und es kam ein normiertes Industrielastprofil zum Einsatz, welches auf Daten des Jahres 2017 basiert. Dieses Lastprofil umfasst die aggregierten Verbrauchswerte sämtlicher Industriezweige. Die Verwendung eines solchen normierten Profils ermöglicht es, eine konsistente und vergleichbare Grundlage für die Analyse des Energieverbrauchs und der Lastverteilung im gesamten deutschen Industriesektor zu schaffen. Diese Simulation zielt darauf ab, ein umfassendes Verständnis für die dynamischen

Veränderungen im Energieverbrauch innerhalb des deutschen Industriesektors zu erlangen, insbesondere im Hinblick auf die zunehmende Integration und Auswirkung der Elektromobilität. Durch die Anwendung spezifischer Modellierungsansätze wurden potenzielle Zukunftsszenarien generiert, die eine fundierte Analyse der erwarteten Entwicklungen ermöglichen und somit wertvolle Einblicke für strategische Entscheidungsprozesse liefern.

4.1.1 Kumulierte Leistungs- und Energiekennzahlen der E-Mobilität

Basierend auf einer durchschnittlichen Ladeleistung von 11 kW/EV und einer durchschnittlichen Batteriekapazität von 60 kWh/EV wird prognostiziert, dass das kumulierte Ladevermögen im deutschen Industriesektor ab dem Jahr 2030 eine Gesamtleistung von 10 GW erreichen wird, welches sich voraussichtlich bis zum Jahr 2045 auf 32 GW erhöhen wird. (Abbildung 28). Für die durchgeführten Simulationen ist insbesondere die kumulierte verfügbare Speicherkapazität von Bedeutung. Es wird erwartet, dass diese ab dem Jahr 2030 bei 17 GWh liegt und bis zum Jahr 2045 auf 53 GWh anwachsen wird, unter der Prämisse, dass 90% der Fahrzeuge reine Elektroantriebe nutzen werden.

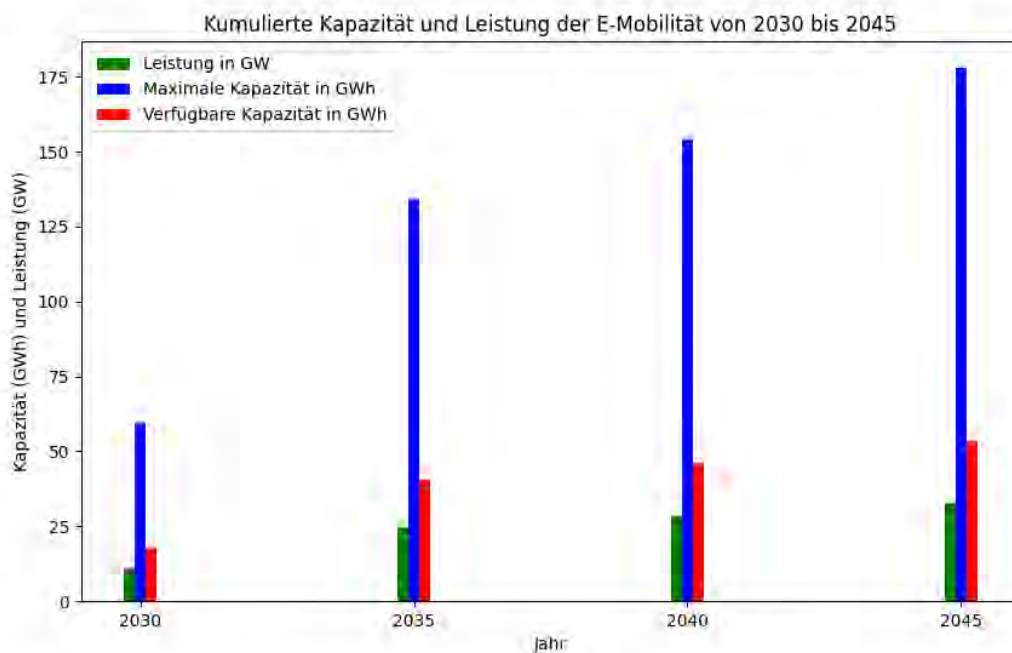


Abbildung 28 - kumulierte Kapazitäts- und Leistungsentwicklung GWh/GW der Entwicklung der E-Mobilität von 2030 bis 2045 (Eigene Darstellung)

In Anbetracht der zeitlichen Variabilität von Batteriespeichersystemen und der Anforderung, dass die Fahrzeuge zum Zeitpunkt der Abfahrt vollständig aufgeladen sein müssen, repräsentiert die kumulierte verfügbare Kapazität eine theoretische Größe. Diese stellt die maximale Energiemenge dar, die unter idealen Bedingungen verfügbar wäre, berücksichtigt jedoch nicht die tatsächlichen Lade- und Entladezyklen sowie die Nutzungsbedingungen im realen Betriebskontext. Daher ist eine Differenzierung zwischen der theoretisch verfügbaren und der tatsächlich nutzbaren Speicherkapazität erforderlich, um eine präzise Einschätzung der Energieverfügbarkeit für Lastmanagementstrategien und für die Sicherstellung der Ladebedürfnisse vor der Abfahrt zu ermöglichen.

4.1.2 Vergleich der verschiedenen Ladeszenarien

Zur Evaluation der Effekte unterschiedlicher Ladestrategien auf das Energienetz wurden innerhalb der Simulation vier distinkte Ladeszenarien implementiert und analysiert. Im ungünstigsten Szenario, in dem sämtliche Elektrofahrzeuge unmittelbar bei ihrer Ankunft aufgeladen werden, resultierte dies – insbesondere während Zeiten hoher Netzbelastung – in signifikanten zusätzlichen Ladespitzen. Wie die Abbildung 29 dargestellt, erhöht sich hier die Spitzenlast um 38 %. Bei rund 56 GW des Industrielastprofil ohne Laden (blaue Kurve), erhöht sich die Leistung auf rund 75 GW bei der Lastprofil mit Laden (orange Kurve).

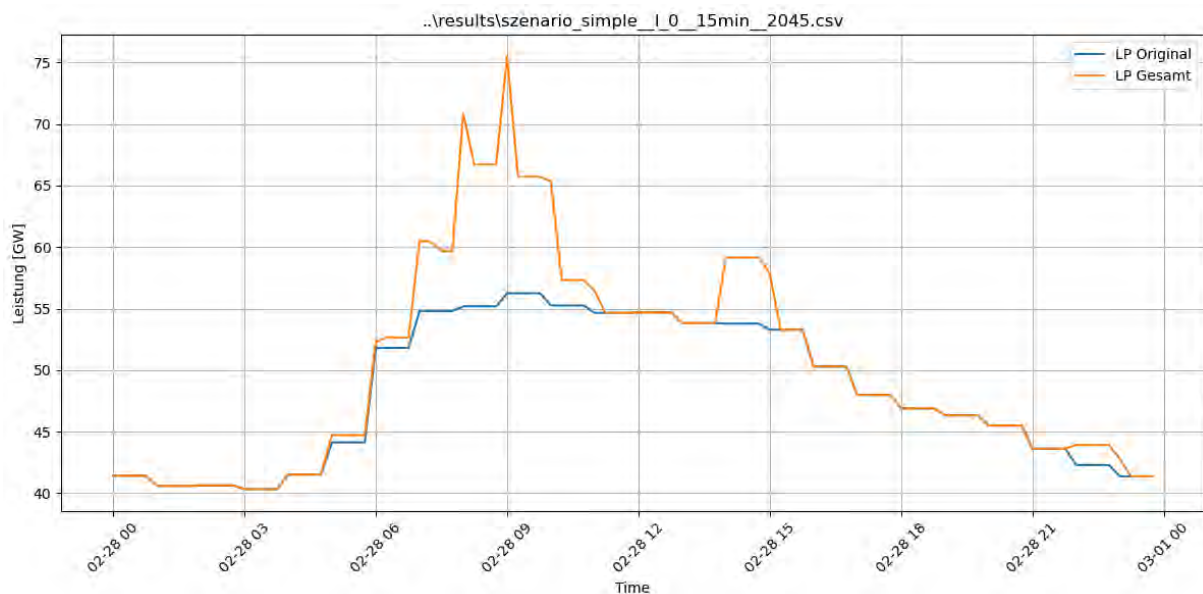


Abbildung 29 - Lastprofil gesamte Industrie Szenario Simple Laden in GW 2045 (Eigene Darstellung)

Demgegenüber tragen die alternativen Szenarien des Ladens mit einer sogenannten „Ladebremse“ in Abbildung 30 und des Ladens „End of Day“ in Abbildung 31 dazu bei, das Auftreten zusätzlicher Spitzenlasten zu verhindern, indem die Ladevorgänge auf Zeiten niedrigerer Netzlast verlagert werden. Im Szenario „Ladebremse“ wurde die Spitzenlast von 56 GW nicht überschritten und alle EVs waren zum Zeitpunkt der Abfahrt geladen. Im Szenario „End of Day“ konnte die Spitzenlast nur auf einen späteren Zeitpunkt verschoben werden und blieb nur minimal unter der des Szenarios „Simple“.

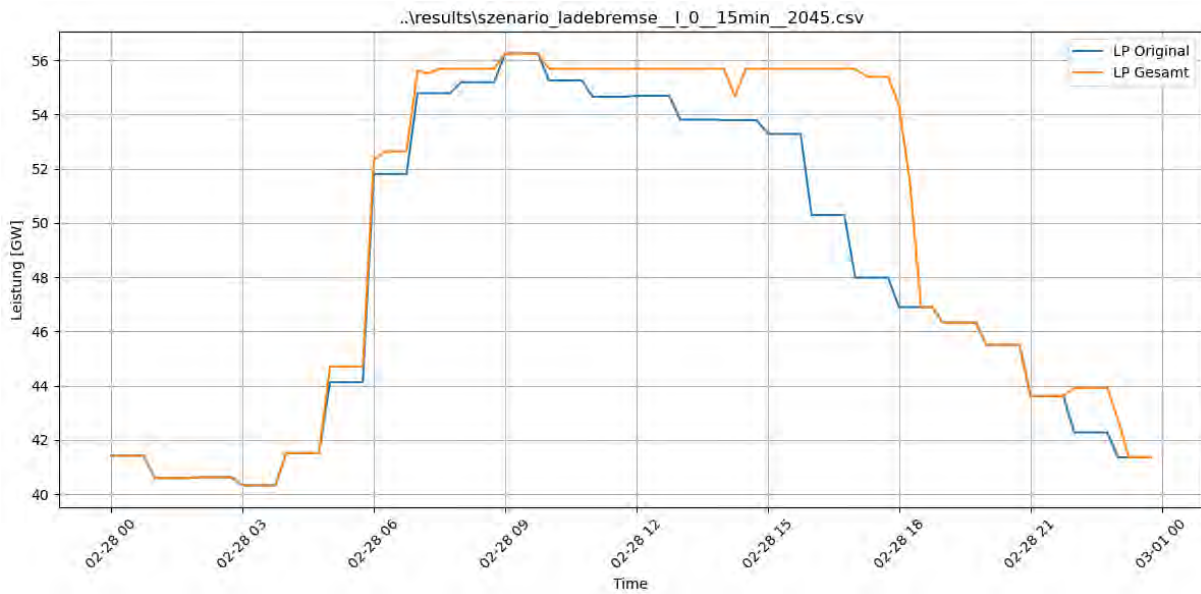


Abbildung 30 - Lastprofil gesamte Industrie Szenario Ladebremse Laden in GW 2045 (Eigene Darstellung)

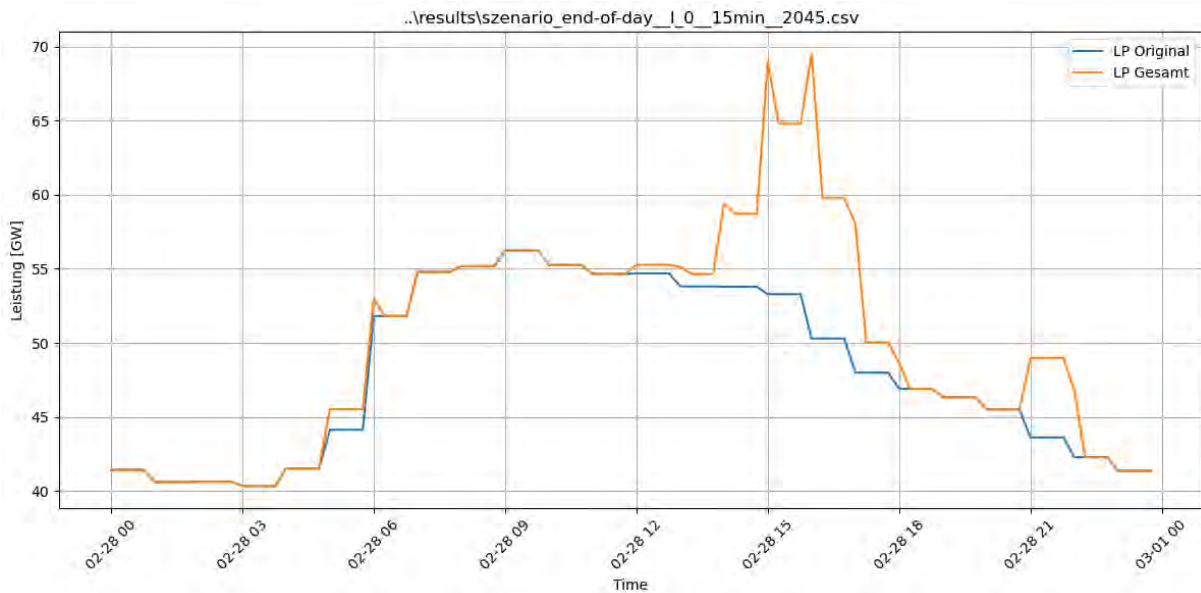


Abbildung 31 - Lastprofil gesamte Industrie Szenario „End of Day“ Laden in GW 2045 (Eigene Darstellung)

Die Implementierung des bidirektionalen Ladens „Bidi“ wie in Abbildung 32, zeigt darüber hinaus das Potenzial, die Spitzenlasten nicht nur zu reduzieren, sondern auch die Ladevorgänge in noch weniger belastete Netzzeiten zu verschieben, was eine effizientere Nutzung der verfügbaren Kapazitäten ermöglicht.

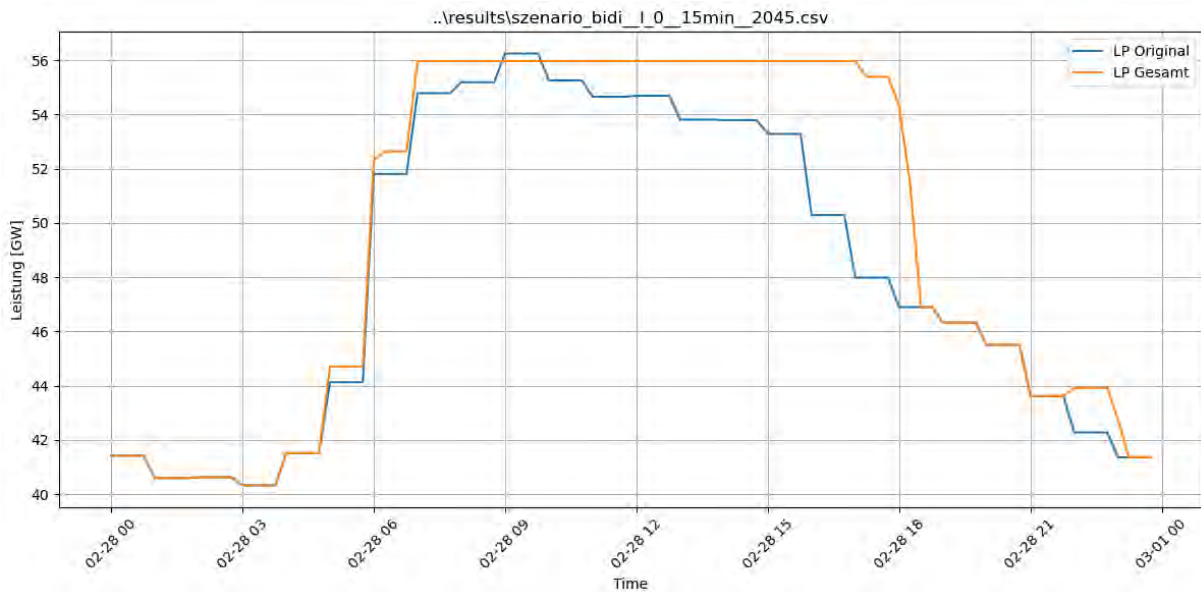


Abbildung 32 - Lastprofil gesamte Industrie Szenario „Bidi“ Laden in GW 2045 (Eigene Darstellung)

In der durchgeführten Simulation in Abbildung 32 wurde festgestellt, dass die kumulierte Spitzenlast der Industrie ohne die Einbeziehung von Ladevorgängen für Elektrofahrzeuge 56,24 GW betrug. Durch die Implementierung von bidirektionalem Laden konnte diese Spitzenlast im Szenario 2045 auf 55,79 GW reduziert werden. Die resultierende Verringerung der kumulierten Spitzenlast beläuft sich somit auf 282 MW.

Die folgende Tabelle 1 zeigt den Anstieg der Ladeszenarien „Simple“ und „Bidi“ über die Jahre 2030 bis 2045 zum HLZ 9 Uhr.

Jahr	HLZ Simple Original (GW)	HLZ Simple Laden (GW)	Anstieg der Spitzenlast (%)	HLZ Bidi Original (GW)	HLZ Bidi Laden (GW)	Anstieg der Spitzenlast (%)
2030	34.6611855310380	41.0734366769114	18	34.6611855310380	34.4878796033828	-0,5
2035	40.6481175773082	55.1825535079546	35	40.6481175773082	40.4448769894216	-0,5
2040	50.1011681766822	66.7730211559531	33	50.1011681766822	49.8506623357987	-0,5
2045	56,2456510662753	75.4824045038955	33	56,2456510662753	55.9644228109439	-0,5

Tabelle 1 - Vergleich Anstieg der Spitzenlasten Gesamtindustrie 2030 bis 2045 (Eigene Darstellung)

Die Abbildung 33 veranschaulicht die Ladeenergie GW, die basierend auf verschiedenen Verfügbarkeitszeiten beladen (positiv) und entladen (negativ) werden konnte.

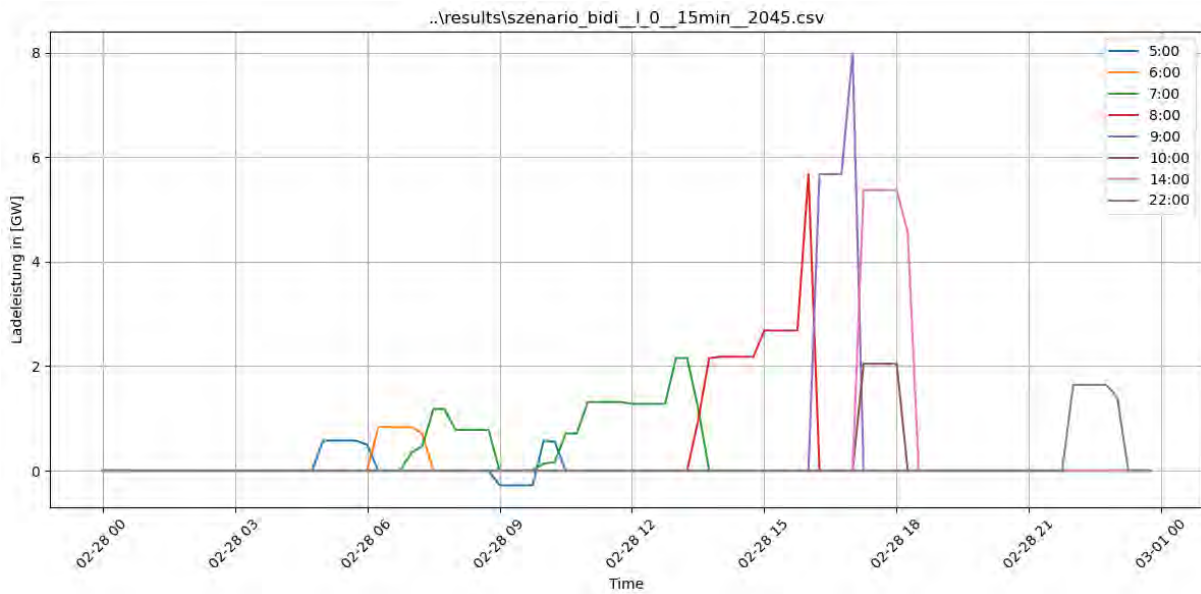


Abbildung 33 – Ladeenergie des bidirektionalen Ladens in GW 2045 (Eigene Darstellung)

Um zu garantieren, dass alle EVs zum Zeitpunkt ihrer Abfahrt vollständig aufgeladen sind, visualisiert die nachstehende Abbildung 34 die Speicherkapazität zu verschiedenen Zeitpunkten. Diese Darstellung beginnt mit einem Startwert, der die Akkukapazität bei der Ankunft – angenommen 80 Prozent – repräsentiert, und endet mit dem Endwert, der die maximale Kapazität der Fahrzeugbatterien darstellt. Diese visuelle Aufbereitung dient dazu, die verfügbare Speicherkapazität im Zeitverlauf zu veranschaulichen, sodass eine effiziente Planung der Ladevorgänge ermöglicht wird, um die vollständige Aufladung der Fahrzeuge vor ihrer geplanten Abfahrt zu sichern.

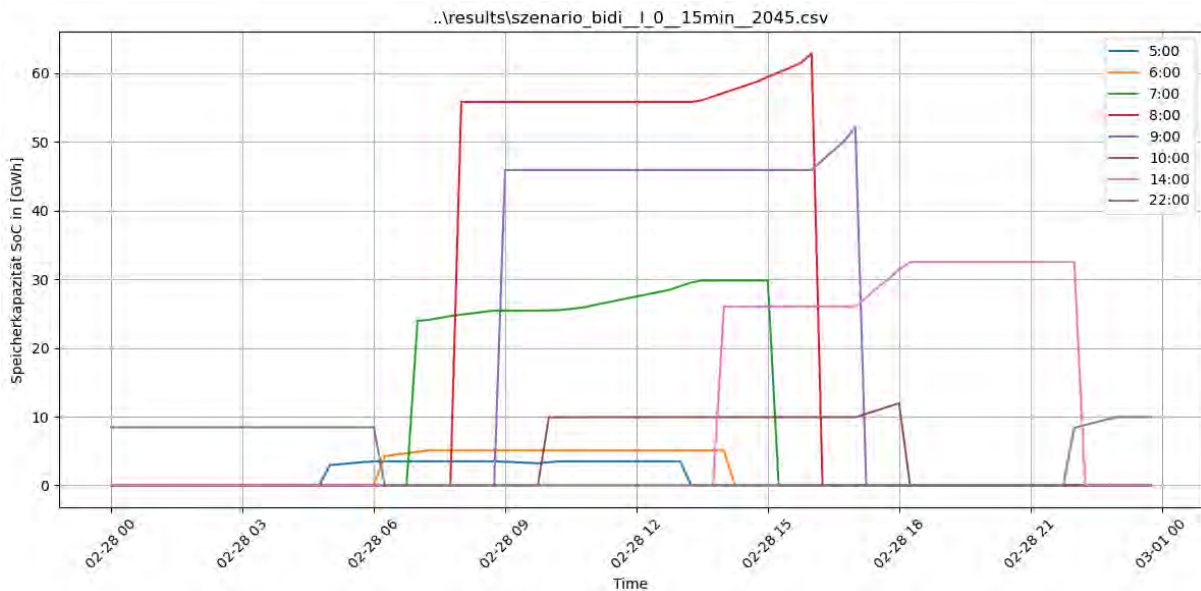


Abbildung 34 – Szenario „Bidi“ Speicherkapazität SoC in GWh 2045 (Eigene Darstellung)

4.2 Ergebnisse der Chemie-Industrie

In Deutschland sind (Stand:16.12.2021) ungefähr 2.100 Unternehmen in der Chemie- und Pharmabranche ansässig, die zusammen etwa 466.500 Personen beschäftigen. Dabei sind 92 Prozent der Chemieunternehmen in Deutschland kleine bis mittelgroße Betriebe mit einer Belegschaft von weniger als 500 Mitarbeitenden. [27] In der energieintensiven Chemieindustrie belief sich der Energieverbrauch im Jahr 2020 auf 42,94 TWh/a. Es wird prognostiziert, dass dieser Verbrauch bis zum Jahr 2045 auf über 67 TWh/a ansteigen wird, wie in der entsprechenden Abbildung 35 dargelegt. [20]

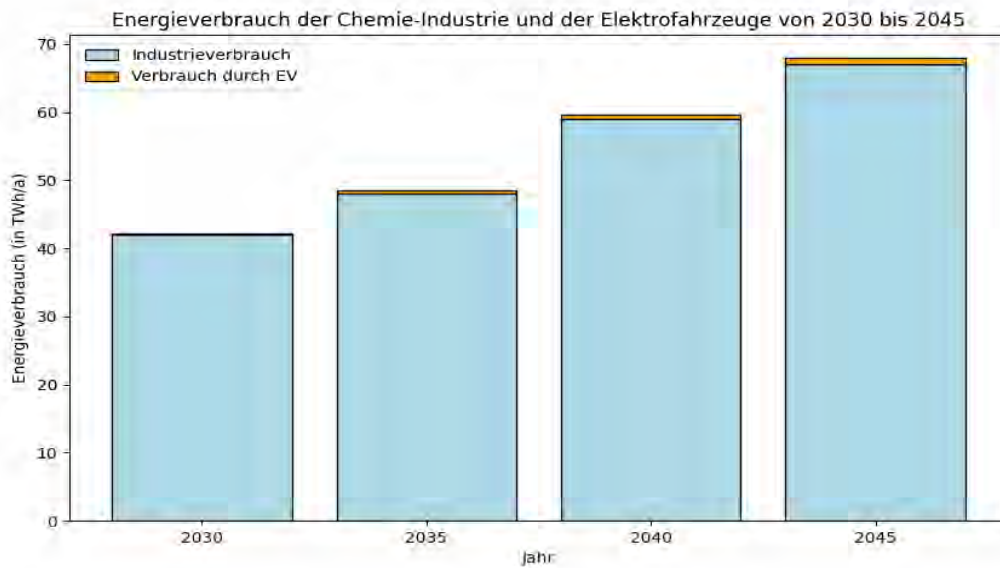


Abbildung 35 - Prognostizierter Energieverbrauch der Chemie-Industrie in TWh/a bis 2045 (Eigene Darstellung) [20]

Das ungünstigste Szenario, das vorsieht, dass alle Elektrofahrzeuge sofort bei ihrer Ankunft geladen werden, führte zu erheblichen zusätzlichen Ladespitzen, besonders in Zeiten hoher Netzbelastung. Wie in Abbildung 36 illustriert, resultiert dies in einem Anstieg der Spitzenlast um 9 %. Bei einer Spitzenlast ohne Laden von etwa 11,8 GW erhöhte sich die Spitzenlast mit Laden auf 12,8 GW.

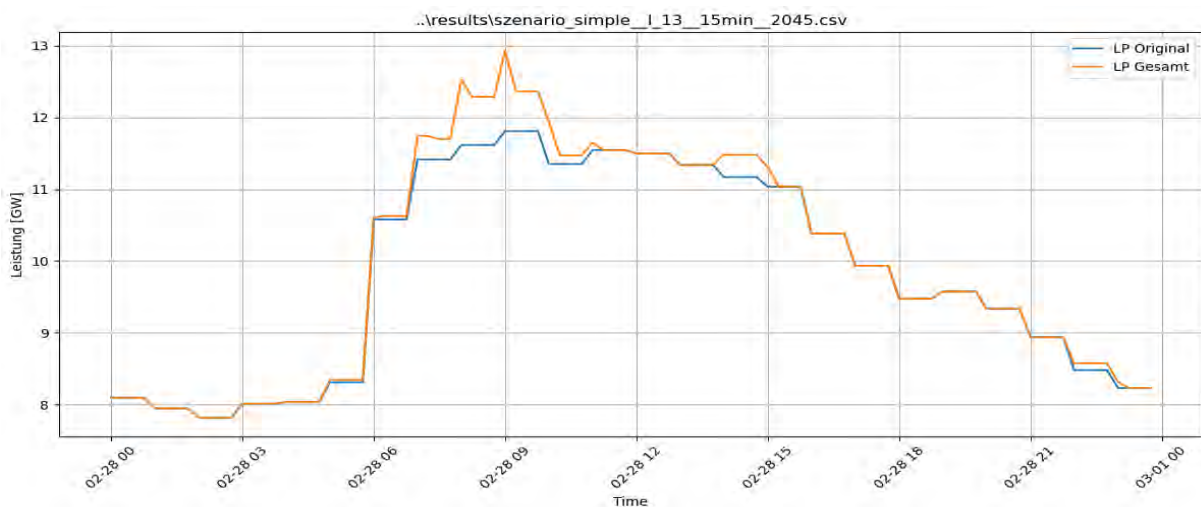


Abbildung 36 - Lastprofil gesamte Chemie-Industrie Szenario Simple Laden in GW 2045 (Eigene Darstellung)

In der spezifischen Simulation für den Chemiesektor ergab sich, dass die kumulierte Spitzenlast ohne Berücksichtigung der Ladevorgänge von Elektrofahrzeugen, 11,81 GW erreichte. Durch die Einführung von bidirektionalem Laden konnte diese Spitzenlast im betrachteten Szenario für das Jahr 2045 auf 11,75 GW gesenkt werden. Die daraus resultierende Reduktion der kumulierten Spitzenlast summiert sich auf 100 MW.(Abbildung 37)

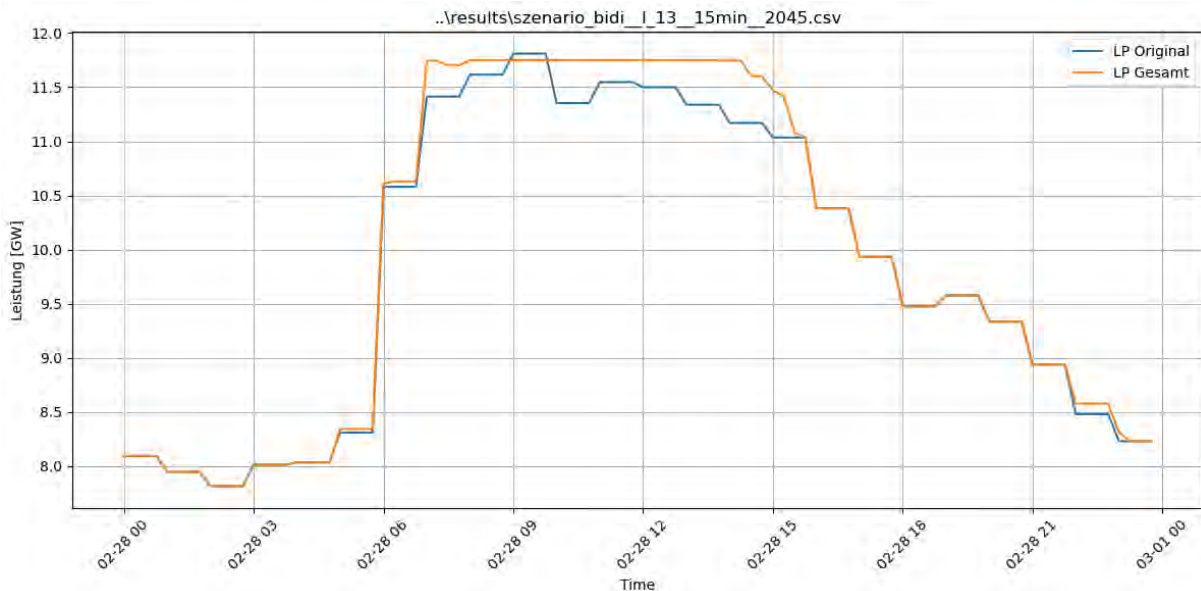


Abbildung 37 - Lastprofil gesamte Chemie-Industrie Szenario „Bidi Laden“ in GW 2045 (Eigene Darstellung)

Die folgende Tabelle 2 zeigt den Anstieg der Ladeszenarien Simple und Bidirektional (Bidi) über den Zeitraum von 2030 bis 2045 zum HLZ 9 Uhr:

Jahr	HLZ Simple Original (GW)	HLZ Simple Laden (GW)	Anstieg der Spitzenlast (%)	HLZ Bidi Original (GW)	HLZ Bidi Laden (GW)	Anstieg der Spitzenlast (%)
2030	7.5427463202806	7.9166607152243	5	7.5427463202806	7.5050325886792	-0,4
2035	8.5685879250882	9.4161272202940	10	8.5685879250882	8.5257449854628	-0,4
2040	10.520497499167	11.492674926021	9	10.520497499167	10.467895011671	-0,4
2045	11.810704106583	12.932447291415	9	11.810704106583	11.751650586050	-0,5

Tabelle 2 - Vergleich Anstieg der Spitzenlasten Chemie-Industrie 2030 bis 2045 (Eigene Darstellung)

Auch im Rahmen dieses Szenarios des bidirektionalen Ladens wurde gewährleistet, dass alle Fahrzeuge zum Zeitpunkt ihrer Abfahrt vollständig aufgeladen waren, wie in Abbildung 38 dargestellt.

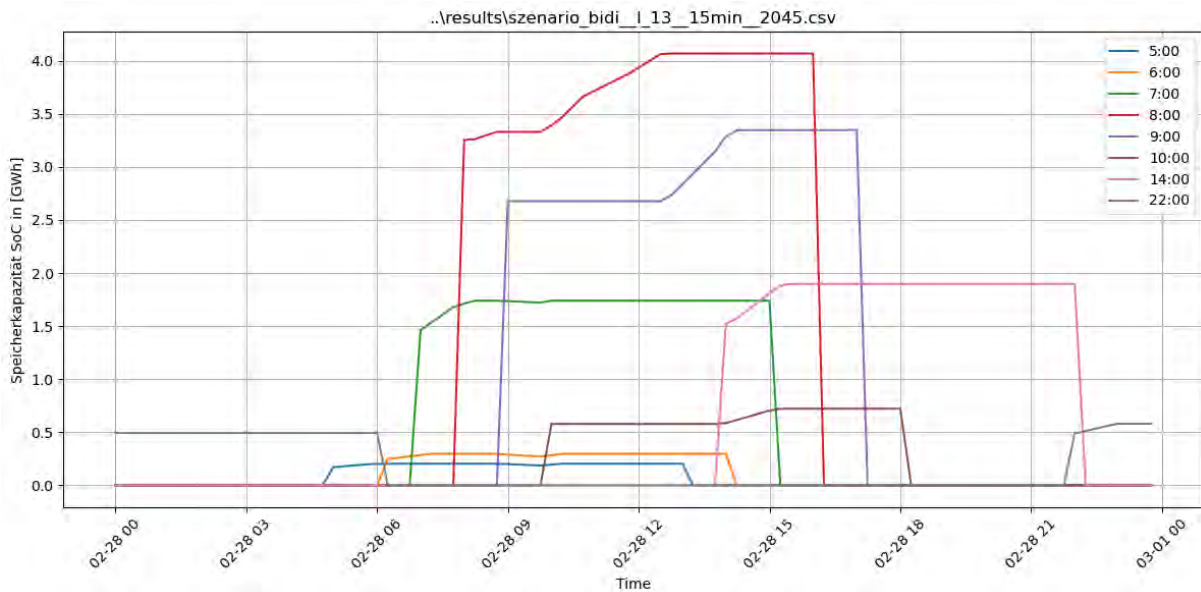


Abbildung 38 - Szenario „Bidi“ Speicherkapazität SoC in GWh 2045 Chemie-Industrie (Eigene Darstellung)

4.3 Ergebnisse der Automobileindustrie

Die Automobilindustrie stellt innerhalb des verarbeitenden Gewerbes die bedeutendste Branche in Deutschland dar und zeichnet sich durch den höchsten Umsatz aller Industriezweige aus. Im Jahr 2021 generierten die Unternehmen dieses Sektors einen Umsatz von über 411 Milliarden Euro und boten direkt nahezu 786.000 Menschen eine Beschäftigung. [28] In der Fahrzeugindustrie betrug der Energieverbrauch im Jahr 2020 10,9 TWh/a. Es wird erwartet, dass dieser Verbrauch bis zum Jahr 2045 auf über 17,06 TWh/a ansteigen wird, wie es in Abbildung 39 dargestellt ist. [20]

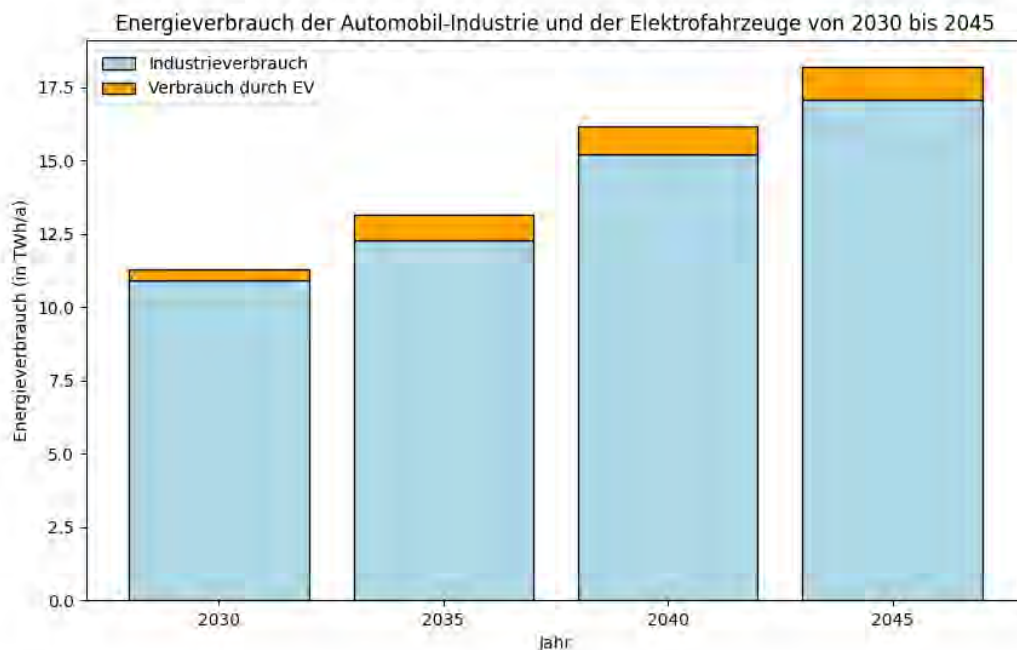


Abbildung 39 - Prognostizierter Energieverbrauch der Automobil-Industrie bis 2045 (Eigene Darstellung) [20]

Das kritischste Szenario, welches annimmt, dass alle Elektrofahrzeuge unmittelbar bei ihrer Ankunft geladen werden, zog deutlich erhöhte Ladespitzen nach sich, insbesondere während Phasen mit hoher Netzbelastung. Wie Abbildung 40 verdeutlicht, bewirkte dies eine Zunahme der Spitzenlast um 41 Prozent. Bei einer Spitzenlast ohne Laden von etwa 2,8 GW erhöhte sich die Spitzenlast mit Laden auf 4,7 GW. (Abbildung 40)

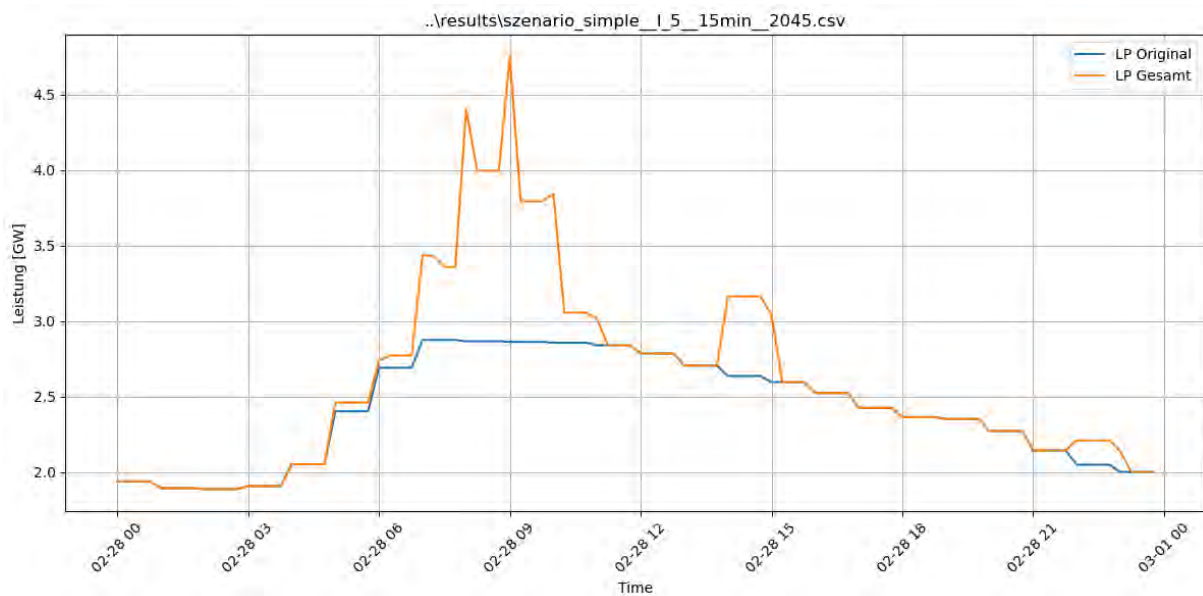


Abbildung 40 - Lastprofil gesamte Automobil-Industrie Szenario „Simple“ in GW 2045 (Eigene Darstellung)

In der gezielten Simulation für die Fahrzeugindustrie wurde festgestellt, dass die kumulierte Spitzenlast, ohne die Ladevorgänge von Elektrofahrzeugen einzurechnen, 2,87 GW betrug. Durch die Implementierung des bidirektionalen Ladens, wie in Abbildung 41 dargestellt, konnte diese Spitzenlast für das Szenario im Jahr 2045 auf 2,86 GW reduziert werden. Die daraus resultierende Verringerung der kumulierten Spitzenlast beläuft sich auf 10 MW.

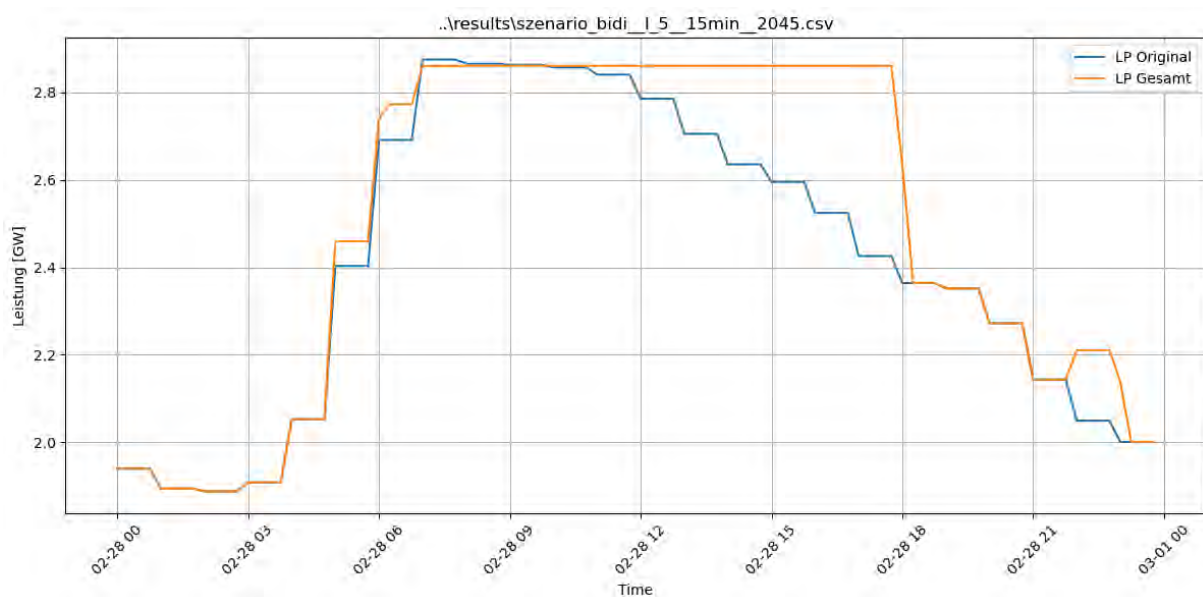


Abbildung 41 - Lastprofil gesamte Automobil-Industrie Szenario „Bidi“ in GW 2045 (Eigene Darstellung)

Die folgende Tabelle 3 zeigt den Anstieg der Ladeszenarien Simple und Bidirektional (Bidi) über den Zeitraum von 2030 bis 2045 zum HLZ 7 Uhr:

Jahr	HLZ Simple Original (GW)	HLZ Simple Laden (GW)	Anstieg der Spitzenlast (%)	HLZ Bidi Original (GW)	HLZ Bidi Laden (GW)	Anstieg der Spitzenlast (%)
2030	1.8372453917616	2.4593302748644	34	1.8372453917616	1.8280591648028	-0,4
2035	2.0732218640980	3.4922943097569	68	2.0732218640980	2.0628557547775	-0,4
2040	2.5620302710804	4.1889971072034	63	2.5620302710804	2.5492201197250	-0,4
2045	2.8755418700416	4.7531588960982	66	2.8755418700416	2.8611641606914	-0,7

Tabelle 3 - Vergleich Anstieg der Spitzenlasten Automobile-Industrie 2030 bis 2045 (Eigene Darstellung)

4.4 Ergebnisse der Metall-Industrie

Im Jahr 2023 verzeichnete der Sektor der Metallerzeugung und -bearbeitung in Deutschland eine Beschäftigtenzahl von etwa 228.000 Personen. Im Vergleich zu anderen Industriezweigen zählt die Metallindustrie zu den energieintensivsten Sektoren. [26] Im Jahr 2020 belief sich der Energieverbrauch in diesem Bereich auf etwa 80 TWh/a. Prognosen zufolge wird erwartet, dass dieser Verbrauch bis zum Jahr 2045 auf über 120 TWh/a ansteigt, wie in Abbildung 42 dargestellt. [20]

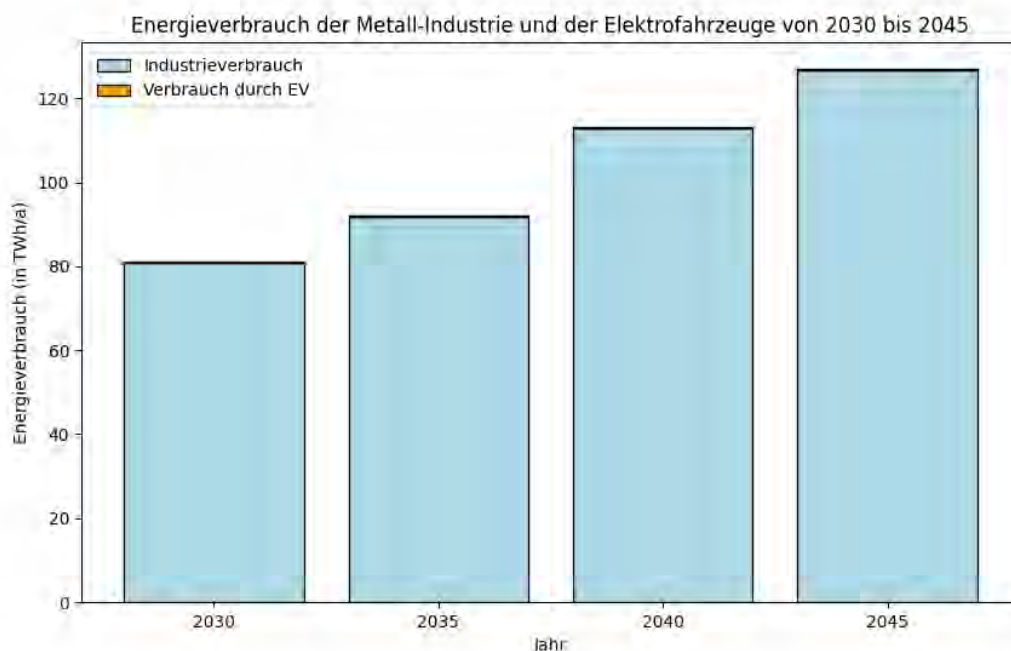


Abbildung 42 – Prognostizierter Energieverbrauch der Metall-Industrie bis 2045 in TWh/a (Eigene Darstellung) [20]

In der Metallindustrie wurde festgestellt, dass die Spitzenlasten, wie in Abbildung 43 gezeigt, während des Hochleistungszeitraums um 6 Uhr morgens nur teilweise reduziert werden konnten, da zu diesem Zeitpunkt zu wenige Fahrzeuge für das bidirektionale Laden zur Verfügung standen. Dies deutet darauf hin, dass die zugrunde gelegten Beschäftigungsdaten, möglicherweise nicht die tatsächlichen Verhältnisse innerhalb der Metallindustrie widerspiegeln. Eine Diskrepanz zwischen den angenommenen und den realen Beschäftigungszahlen könnte eine Erklärung für die unzureichende Verfügbarkeit von Fahrzeugen sein, die erforderlich wäre, um effektive Maßnahmen zur Lastspitzenreduktion durchzuführen, wie in Abbildung 44 illustriert.

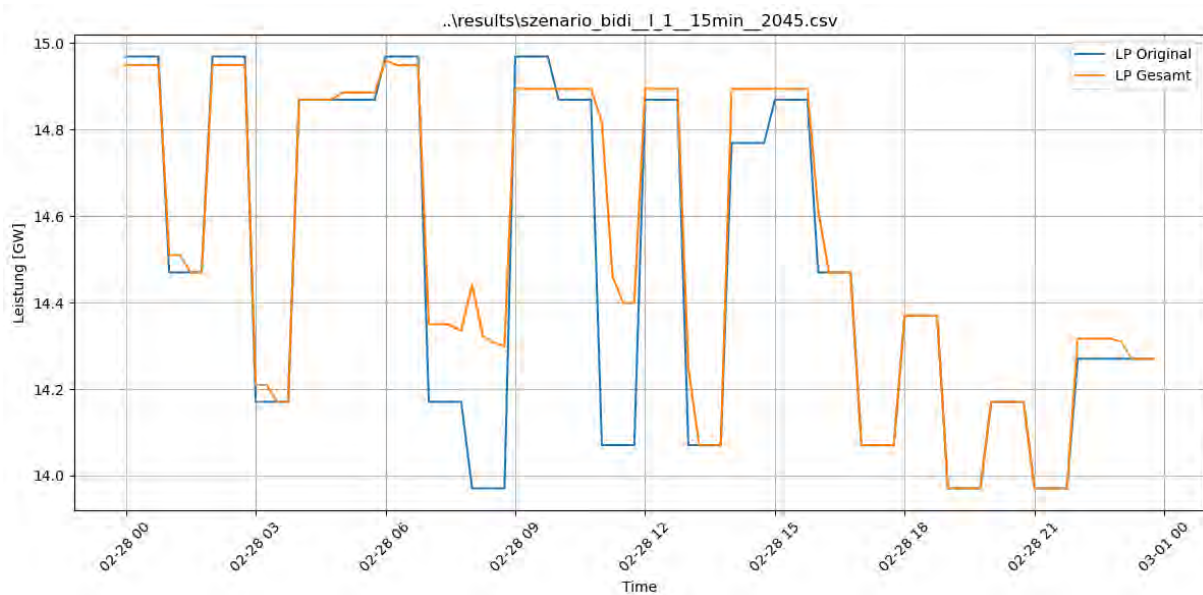


Abbildung 43 - Lastprofil gesamte Metall-Industrie Szenario „Bidi“ in GW 2045 (Eigene Darstellung)

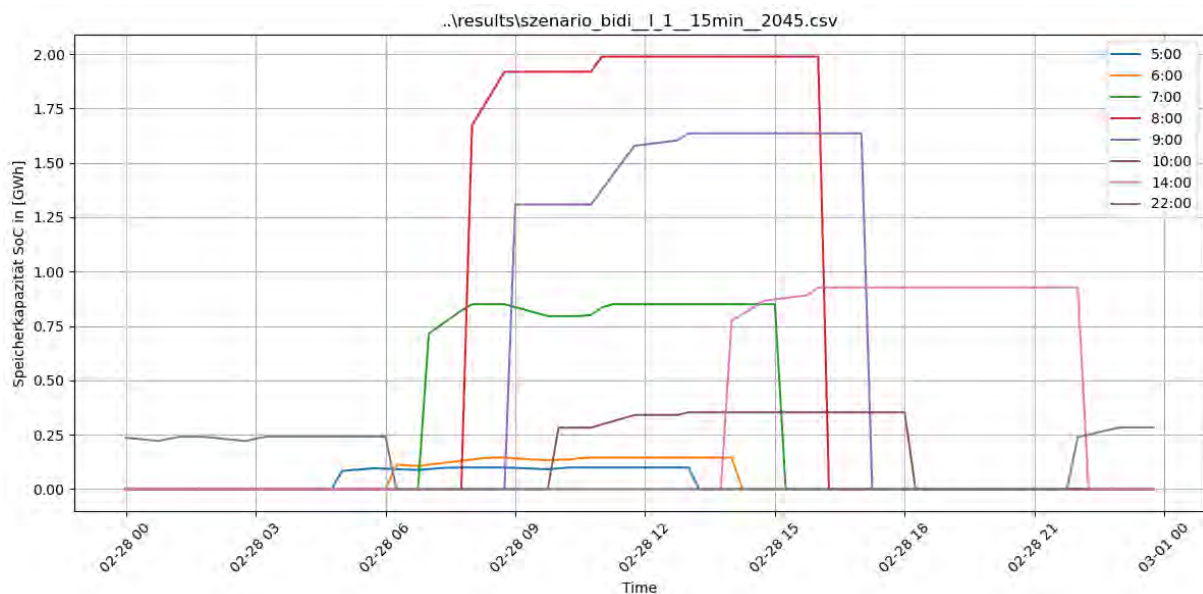


Abbildung 44 - Szenario „Bidi“ Speicherkapazität SoC in GWh 2045 Metall-Industrie (Eigene Darstellung)

Die folgende Tabelle 4 zeigt den Anstieg der Ladeszenarien Simple und Bidirektional (Bidi) über den Zeitraum von 2030 bis 2045 zur HLZ 6 Uhr:

Jahr	HLZ Simple Original (GW)	HLZ Simple Laden (GW)	Anstieg der Spitzenlast (%)	HLZ Bidi Original (GW)	HLZ Bidi Laden (GW)	Anstieg der Spitzenlast (%)
2030	9.5597378287485	9.5643776042524	0,04	9.5597378287485	9.5570036753266	-0,03
2035	10.859096732940	10.869613557415	0,06	10.859096732940	10.852899318517	-0,03
2040	13.333784827741	13.345848244051	0,06	13.333784827741	13.326676028844	-0,03
2045	14.968614576288	14.982533902799	0,06	14.968614576288	14.960412116022	-0,03

Tabelle 4 - Vergleich Anstieg der Spitzenlasten Metall-Industrie 2030 bis 2045 (Eigene Darstellung)

4.5 Ergebnisse Maschinenbau-Industrie

Im Januar 2023 zählte der Maschinenbausektor in Deutschland ungefähr 956.244 Beschäftigte. [26] In der Maschinenbauindustrie belief sich der Energieverbrauch im Jahr 2020 auf 6,46 Terawattstunden

pro Jahr (TWh/a). Prognosen zufolge wird erwartet, dass dieser Verbrauch bis zum Jahr 2045 auf über 10,11 TWh/a anwachsen wird, wie in Abbildung 45 illustriert. [20]

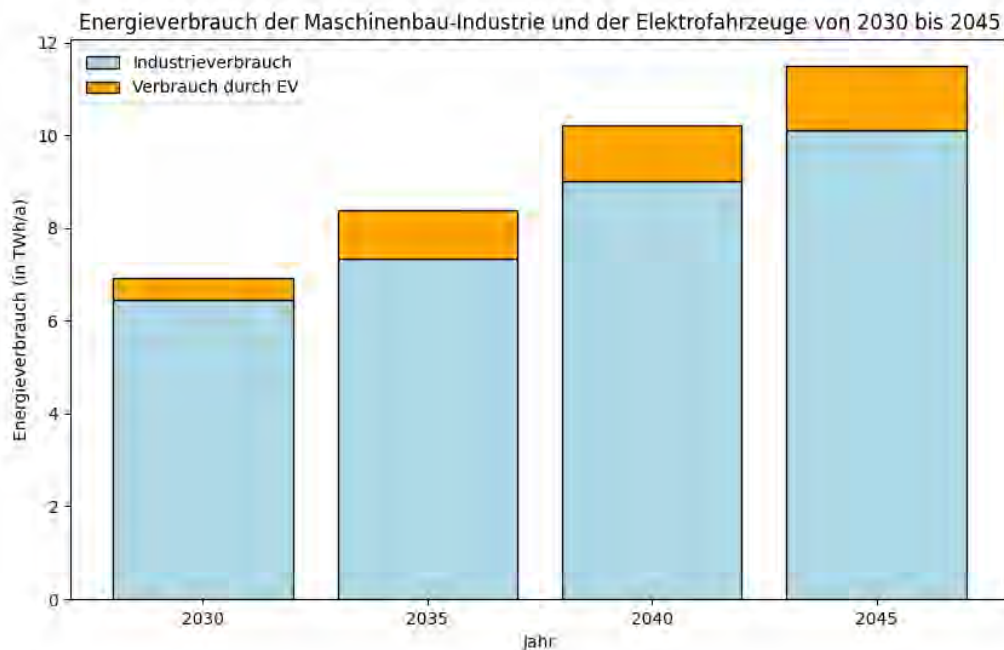


Abbildung 45 - Prognostizierter Energieverbrauch der Maschinenbau-Industrie bis 2045 in TWh/a (Eigene Darstellung) [20]

Das Szenario „Simple“, welches annimmt, dass alle Elektrofahrzeuge unmittelbar bei ihrer Ankunft geladen werden, zog deutlich erhöhte Ladespitzen nach sich, insbesondere während Phasen mit hoher Netzbelastung. Wie verdeutlicht in Abbildung 46, bewirkte dies eine enorme Zunahme der Spitzenlast. Bei einer Spitzenlast ohne Laden von etwa 1,7 GW erhöhte sich die Spitzenlast mit Laden auf 4 GW.

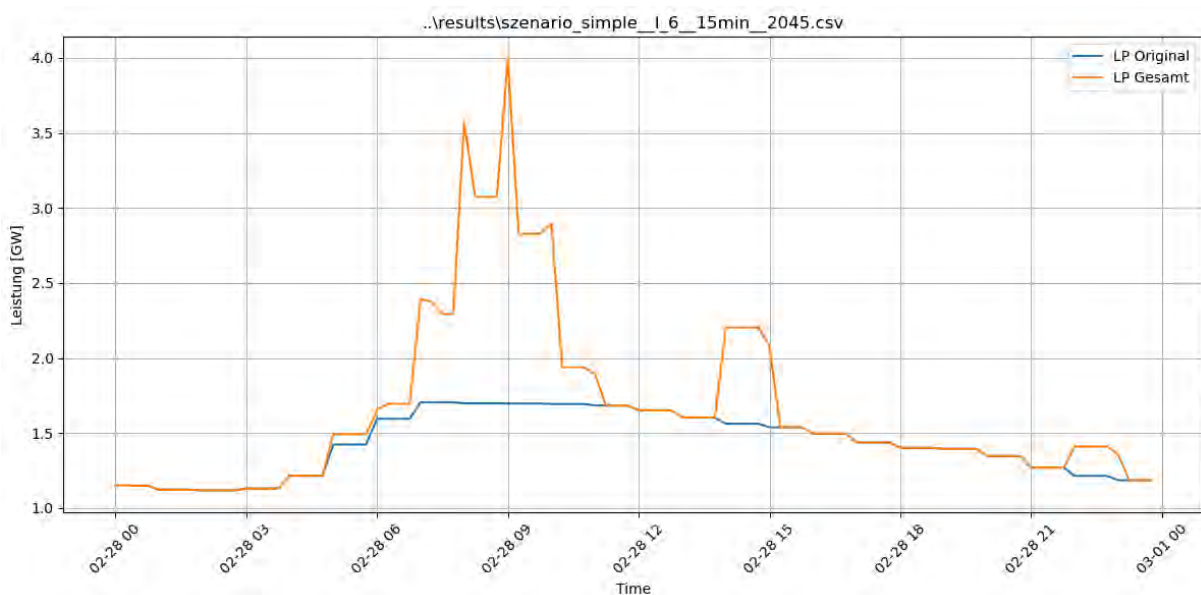


Abbildung 46 - Lastprofil gesamte Maschinenbau-Industrie Szenario „Simple“ in GW 2045 (Eigene Darstellung)

Im Szenario „Ladebremse“ in Abbildung 47 wurde die Spitzenlast von 56 GW nicht überschritten und alle EVs waren zum Zeitpunkt der Abfahrt geladen.

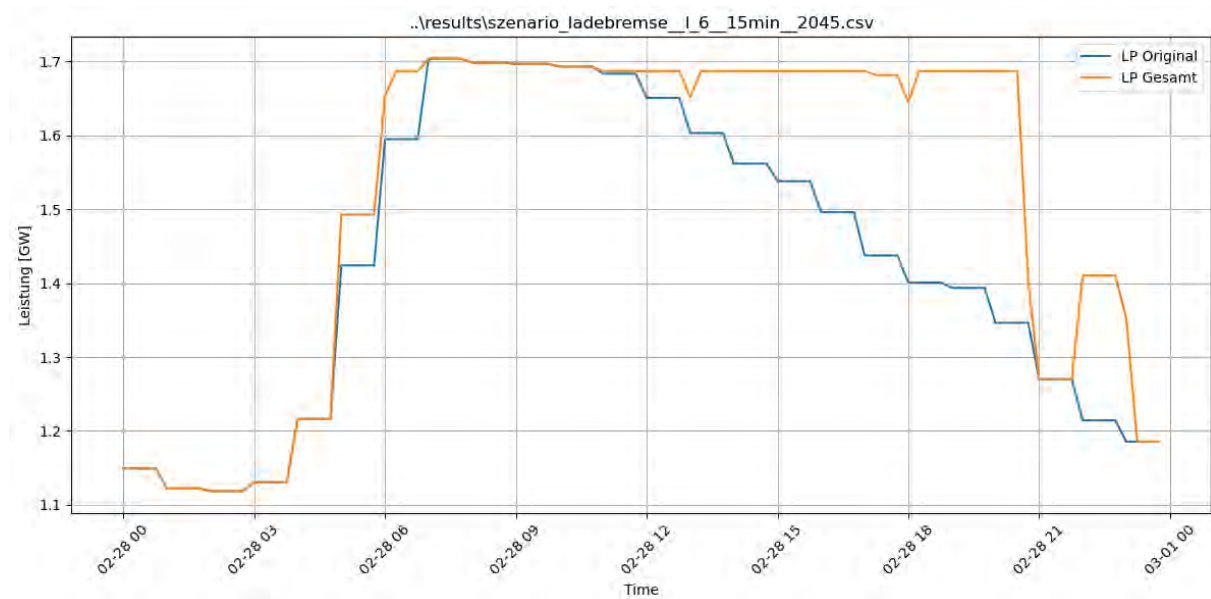


Abbildung 47 - Lastprofil Maschinenbau-Industrie Szenario Ladebremse Ladenin GW 2045 (Eigene Darstellung)

In der spezifischen Simulation für den Maschinenbausektor ergab sich, dass die kumulierte Spitzenlast ohne Berücksichtigung der Ladevorgänge für Elektrofahrzeuge 1,7 GW erreicht. Wie in Abbildung 48 aufgezeigt, konnte durch die Einführung des bidirektionalen Ladens diese Spitzenlast im Szenario für das Jahr 2045 auf 1,69 GW gesenkt werden. Die daraus resultierende Reduktion der kumulierten Spitzenlast summiert sich auf 12,58 MW.

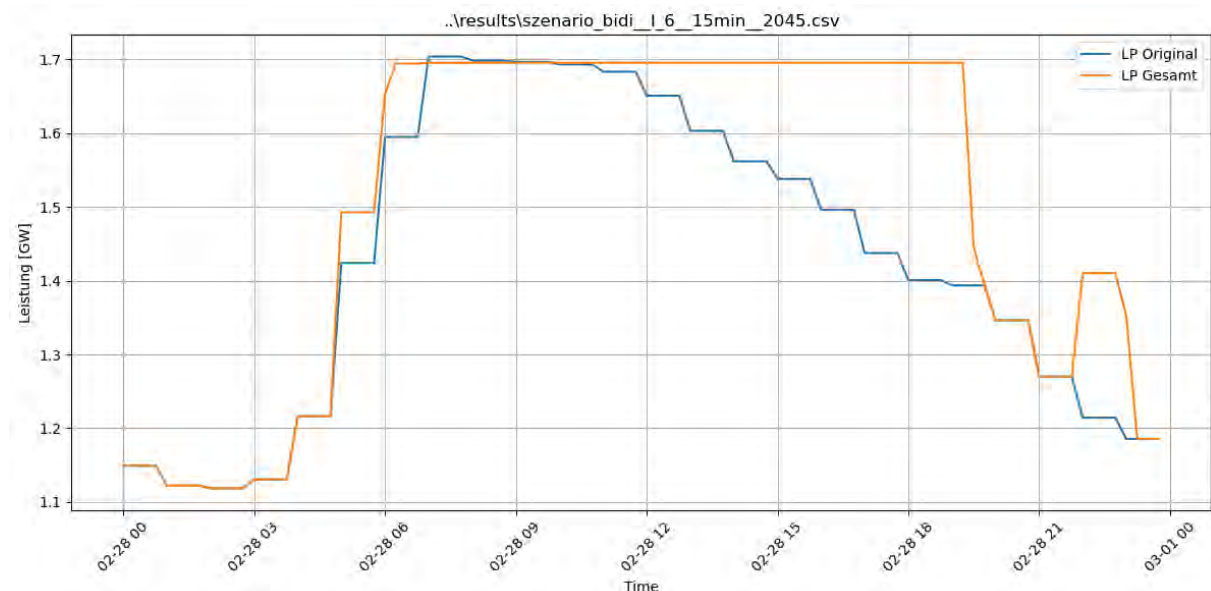


Abbildung 48 - Lastprofil gesamte Maschinenbau-Industrie Szenario „Bidi“ in GW 2045 (Eigene Darstellung)

Die folgende Tabelle 5 zeigt den Anstieg der Ladeszenarien Simple und Bidirektional (Bidi) über den Zeitraum von 2030 bis 2045 zum HLZ 7 Uhr:

Jahr	HLZ Simple Original (GW)	HLZ Simple Laden (GW)	Anstieg der Spitzenlast (%)	HLZ Bidi Original (GW)	HLZ Bidi Laden (GW)	Anstieg Spitzenlast (%)
2030	1,0888628652066	1.8504337215276	71	1,0888628652066	1,0847164655801	-0,4
2035	1,2371909335320	2.9687235563179	140	1,2371909335320	1,2277924277119	-0,8
2040	1,5171601624961	3.5044178680906	131	1,5171601624961	1,5063795234671	-0,72
2045	1,7040872395107	3.9955344141258	134	1,7040872395107	1,6916480406311	-0,74

Tabelle 5 – Vergleich Anstieg der Spitzenlasten Maschinenbau - Industrie 2030 bis 2045 (Eigene Darstellung)

5 Diskussion und Ausblick

Die Diskussion der vorliegenden Arbeit widmet sich einer umfassenden Bewertung der gewonnenen Erkenntnisse im Kontext der Implementierung und Nutzung bidirektionaler Ladetechnologien an Industriestandorten. Durch die Analyse verschiedener Ladeszenarien und deren potenzielle Auswirkungen auf das Energieversorgungssystem hat diese Studie sowohl das transformative Potenzial als auch die Herausforderungen dieser Technologie hervorgehoben.

Es wird kritisch reflektiert, dass die zugrunde liegende Berechnungsmethodik auf einem Lastprofil basiert, welches den Energieverbrauch pro Stunde repräsentiert. Dieser Ansatz aggregiert die Leistungswerte in Sekundenintervallen zu einem stündlichen Gesamtverbrauch. Eine solche kumulative Betrachtung birgt jedoch die Limitation, dass temporäre Spitzenlasten, die auf Sekundenbasis auftreten, in der stündlichen Durchschnittsbilanzierung potenziell unterschätzt werden. Die Folge ist, dass die tatsächliche maximale Lastspitze pro Sekunde deutlich über dem durchschnittlichen Energieverbrauch liegen kann, der über eine Stunde hinweg ermittelt wird. Diese Diskrepanz zwischen der kurzfristigen Lastspitze und dem bilanzierten Stundenmittelwert unterstreicht die Notwendigkeit einer differenzierteren Betrachtungsweise bei der Analyse und Bewertung von Lastprofilen, insbesondere im Kontext der Netzstabilität und der Auslegung von Energieversorgungssystemen.

Eines der zentralen Ergebnisse dieser Arbeit ist die Identifikation bidirektionaler Ladetechnologien als einen vielversprechenden Ansatz zur Stabilisierung des Stromnetzes. Insbesondere die Fähigkeit, Elektrofahrzeugbatterien als flexible Speicherressourcen zu nutzen, eröffnet neue Wege für ein effizienteres Energiemanagement.

Der Vergleich der verschiedenen Industriezweige im Kontext der Hochlastzeiten (HLZ) offenbarte, dass Spitzenlasten durch gezielte Maßnahmen reduziert werden konnten. In der energieintensiven Metallindustrie jedoch ließ sich keine Senkung der Spitzenlasten feststellen, da zum kritischen Zeitpunkt um 6 Uhr morgens in der Simulation nicht ausreichend Speicherkapazitäten zur Verfügung standen. Diese Beobachtung legt nahe, dass die herangezogenen Beschäftigungsdaten nicht mit den tatsächlichen Beschäftigungszahlen korrelieren. Eine mögliche Diskrepanz zwischen simulierten und realen Daten kann die Wirksamkeit von Lastmanagementmaßnahmen in der Praxis erheblich beeinflussen.

Die Simulation verschiedener Ladeszenarien verdeutlicht die Bedeutung einer strategischen Planung und Steuerung von Ladevorgängen. Die Szenarien „Laden mit Ladebremse“ und „Laden End of Day“ demonstrieren, wie durch zeitlich angepasste Ladevorgänge die Belastung des Stromnetzes reduziert und die Ladelaast in Zeiten geringerer Nachfrage verschoben werden kann. Das bidirektionale Laden geht noch einen Schritt weiter, indem es nicht nur Lastspitzen vermeidet, sondern aktiv zur Netzstabilisierung beiträgt, indem es Energie zurück ins Netz speist.

Trotz der vielversprechenden Potenziale stehen der breiten Implementierung bidirektionaler Ladetechnologien noch Herausforderungen gegenüber. Dazu gehören technische Limitationen, wie die Notwendigkeit zur Weiterentwicklung der Batterietechnologie und Infrastruktur, sowie regulatorische

und wirtschaftliche Barrieren. Die Akzeptanz und das Engagement der Industrieunternehmen, Energieversorger und politischen Entscheidungsträger sind entscheidend, um diese Herausforderungen zu überwinden.

Ein weiterer diskussionswürdiger Aspekt ist die Rolle der Verbraucher. Die Bereitschaft der Fahrzeugnutzer, ihre Fahrzeuge als Teil eines größeren Energieversorgungssystems einzusetzen, ist eine wichtige Voraussetzung für den Erfolg bidirektionaler Ladetechnologien. Informations- und Bildungsinitiativen könnten dazu beitragen, das Bewusstsein zu schärfen und die Akzeptanz zu fördern.

Abschließend legt diese Arbeit nahe, dass bidirektionales Laden eine Schlüsseltechnologie für die Energiewende darstellt. Die erfolgreiche Integration dieser Technologie erfordert jedoch eine holistische Herangehensweise, die technische Innovation, regulatorische Rahmenbedingungen und die Förderung der Akzeptanz bei allen Stakeholdern umfasst. Die vorliegende Arbeit trägt dazu bei, das Verständnis für die Komplexität und das Potenzial bidirektionaler Ladetechnologien zu erweitern und bietet einen Ausgangspunkt für weiterführende Forschungen in diesem dynamischen Forschungsfeld.

Die Arbeit eröffnet vielfältige Perspektiven für zukünftige Forschungsarbeiten und die praktische Umsetzung im Bereich der bidirektionalen Ladetechnologien und Elektromobilität. Angesichts der zunehmenden Integration von erneuerbaren Energien und der fortschreitenden Elektrifizierung des Verkehrssektors, erweisen sich bidirektionale Ladetechnologien als wesentlich für die Erreichung einer nachhaltigen und effizienten Energiezukunft.

An diesem Punkt offenbart sich eine fundamentale Limitation der durchgeführten Simulation. Um eine Simulation zu generieren, die die Realität präziser widerspiegelt, wäre es erforderlich, ein reales Leistungsprofil auf Sekundenbasis zu kalkulieren, welches die effektive Leistung exakt darstellt. Ein solcher Ansatz würde die Dynamik des Energieverbrauchs und der Lastspitzen mit höherer Genauigkeit abbilden, indem temporäre Fluktuationen und kurzfristige Belastungsspitzen des Stromnetzes detailliert erfasst werden. Die Erstellung eines derartigen Modells, basierend auf Echtzeitdaten, stellt somit ein vielversprechendes Feld für zukünftige Forschungsarbeiten dar. In diesen könnten die Grenzen der aktuellen Simulationsansätze überwunden und ein tiefergehendes Verständnis für die Interaktion zwischen Elektrofahrzeugen und dem Energieversorgungssystem erlangt werden. Die Nutzung von Echtzeitdaten würde es ermöglichen, die Auswirkungen von Ladevorgängen auf das Stromnetz in Echtzeit zu analysieren und somit fundierte Strategien für das Energiemanagement und die Netzstabilisierung zu entwickeln.

Zukünftige Forschungsarbeiten könnten sich vertieft mit der Weiterentwicklung und Optimierung der bidirektionalen Ladetechnologie befassen, um deren Effizienz und Kompatibilität mit unterschiedlichen Fahrzeugtypen und Energiemanagementsystemen zu verbessern. Darüber hinaus ist eine umfassende Analyse der wirtschaftlichen und ökologischen Auswirkungen dieser Technologien vonnöten, um deren langfristige Vorteile und mögliche Herausforderungen besser zu verstehen.

Ein weiterer wichtiger Forschungsbereich betrifft die regulatorischen und infrastrukturellen Rahmenbedingungen. Es gilt, politische Strategien und regulatorische Anreize zu untersuchen, die die Einführung von bidirektionalen Ladestationen fördern und gleichzeitig die Stabilität und Sicherheit des

Stromnetzes gewährleisten. Zudem ist die Entwicklung von Standards für die Interoperabilität zwischen Fahrzeugen, Ladestationen und dem Stromnetz essenziell, um eine breite Marktakzeptanz zu erreichen.

Darüber hinaus sollte der Fokus auf die Integration von bidirektionalen Ladetechnologien in bestehende und zukünftige, urbane wie industrielle Energiekonzepte gelegt werden. Dies umfasst die Untersuchung von Smart-Grid-Lösungen und dezentralen Energieerzeugungssystemen, die eine effiziente Nutzung erneuerbarer Energien ermöglichen und die Netzstabilität verbessern.

Letztlich eröffnet die zunehmende Digitalisierung des Energie- und Mobilitätssektors neue Möglichkeiten für innovative Geschäftsmodelle und Dienstleistungen, die auf bidirektionalem Laden basieren. Die Erforschung dieser Potenziale könnte signifikant zur Beschleunigung des Übergangs zu einer nachhaltigen und elektrifizierten Mobilitäts- und Energieinfrastruktur beitragen.

Zusammenfassend bietet die vorliegende Arbeit einen soliden Ausgangspunkt für zukünftige Untersuchungen in diesem spannenden Forschungsfeld und trägt zur fortlaufenden Diskussion bei, wie bidirektionale Ladetechnologien effektiv implementiert und genutzt werden können, um die Energie- und Verkehrswende voranzutreiben.

6 Fazit

Die vorliegende Untersuchung hat sich eingehend mit der Thematik der bidirektionalen Ladeinfrastrukturen an Industriestandorten auseinandergesetzt und deren Potenzial zur Optimierung des Netzverbrauchs und zur Reduzierung von Spitzenlasten im Stromnetz untersucht. Durch die Entwicklung und Anwendung eines Simulationsmodells wurden verschiedene Ladeszenarien simuliert, analysiert und verglichen, um die Auswirkungen auf das Energienetz zu bewerten.

Die Ergebnisse der Simulationen zeigen deutlich, dass bidirektionale Ladetechnologien eine Schlüsselrolle in der künftigen Energieinfrastruktur spielen können. Insbesondere die Fähigkeit, Elektrofahrzeugbatterien als temporäre Energiespeicher zu nutzen, bietet erhebliche Vorteile für das Lastmanagement und die Stabilisierung des Stromnetzes. Die Szenarien-Analyse verdeutlichte, dass Strategien wie das Laden mit „Ladebremse“ und „End of Day“ sowie insbesondere das bidirektionale Laden dazu beitragen können, Ladespitzen effektiv zu vermeiden und die Ladelast auf Zeiten geringerer Netzbelastung zu verschieben. Bei der Analyse der gesamten Industrie führte das bidirektionale Laden zu einer Verringerung der gesamten Spitzenlast um 282 MW im HLZ 9 Uhr bis 10 Uhr. Diese Reduktion ist vergleichbar mit der Leistung einiger Spitzenlastkraftwerke oder der von 282 Windkraftanlagen.

Um den Herausforderungen extremer Spitzenlasten zu begegnen, die in der Simulation möglicherweise nicht vollständig abgebildet werden können, empfiehlt sich eine Erhöhung der verfügbaren Leistung beim bidirektionalen Laden. Dies ist besonders relevant für die Rückspeisung von Energie ins Netz, die durch die Umwandlung der Ladeinfrastruktur von Gleichstrom (DC) zu Wechselstrom (AC) derzeit auf 11 kW überwiegend limitiert ist. Eine Erhöhung dieser Leistungsgrenze würde es ermöglichen, dass Elektrofahrzeugbatterien zu Zeiten hoher Netzbelastung eine adäquate

Kompensation bieten können. Eine solche Anpassung der Infrastruktur würde nicht nur zur Stabilisierung des Stromnetzes beitragen, sondern auch die Effizienz und Wirksamkeit des bidirektionalen Ladens als Instrument zur Netzunterstützung signifikant verbessern.

Ausgehend von der anfänglichen Hypothese, dass die Elektromobilität die Energieversorgungsnetze zusätzlich belastet, konnte durch die Untersuchung des bidirektionalen Ladens eine Reduktion der Spitzenlast nachgewiesen werden. Auch wenn dieser Effekt als gering erscheinen mag, stellt er einen bedeutenden Befund dar, der die potenzielle Rolle der Elektromobilität bei der Netzstabilisierung und der effizienteren Nutzung der Energieinfrastruktur unterstreicht.

Darüber hinaus wurde ersichtlich, dass die Implementierung von bidirektionalem Laden nicht nur zur Netzstabilisierung beiträgt, sondern auch wirtschaftliche Vorteile mit sich bringt, indem es Energiekosten reduziert. Diese Erkenntnisse unterstreichen die Notwendigkeit, regulatorische und technische Rahmenbedingungen weiterzuentwickeln, um die breite Einführung und Nutzung von bidirektionalen Ladetechnologien zu fördern.

Die Arbeit hat ebenfalls aufgezeigt, dass die erfolgreiche Integration von bidirektionalem Laden in den Industriestandorten eine enge Zusammenarbeit zwischen Industrieunternehmen, Energieversorgern und politischen Entscheidungsträgern erfordert. Die Schaffung von Anreizen und die Entwicklung von Normen und Standards sind entscheidende Schritte, um die Vorteile dieser Technologie voll auszuschöpfen.

Abschließend lässt sich festhalten, dass die vorliegende Arbeit wichtige Einblicke in die Potenziale und Herausforderungen der Nutzung von bidirektionalen Ladesystemen bietet. Sie leistet einen wertvollen Beitrag zur Diskussion über die Zukunft der Elektromobilität und der Energieversorgung und legt den Grundstein für weiterführende Forschungen in diesem dynamischen und zukunftsweisenden Bereich.

7 Literaturverzeichnis

- [1] Bundesregierung, „Klimaschutzprogramm 2030 der Bundesregierung,“ Bundes, 04.10.2023.
- [2] Die Bundesregierung, „Die Bundesregierung, Energie Und Klimaschutz,“ 25 April 2023. [Online]. Available: <https://www.bundesregierung.de/breg-de/schwerpunkte/klimaschutz/energie-wende-beschleunigen-2040310#:~:text=Um%20die%20Klimaschutzziele%20zu%20erreichen,Energien%20und%20f%C3%BCr%20mehr%20Energieeffizienz.> [Zugriff am 1 Dezember 2023].
- [3] Die Bundesregierung, „Bundesregierung , Energie und Klima,“ Netzausbau, 29 Juli 2022. [Online]. Available: <https://www.bundesregierung.de/breg-de/schwerpunkte/klimaschutz/aenderung-energie-wirtschaftsrecht-2023772>. [Zugriff am 01 Dezember 2023].
- [4] Bundesregierung, „Energie und Klimaschutz,“ Nachhaltige Mobilität, 23 Dezember 2022. [Online]. Available: <https://www.bundesregierung.de/breg-de/schwerpunkte/klimaschutz/nachhaltige-mobilitaet-2044132#:~:text=Die%20Art%2C%20wie%20wir%20von,Bahn%20und%20den%20%C3%B6ffentlichen%20Personenverkehr.> [Zugriff am 01 Dezember 2023].
- [5] W. F. C. L. Luise Bangert, „Agora Verkehrswende (2023): Gesteuertes Laden. Warum es sich lohnt, beim Laden von Elektrofahrzeugen auf Stromangebot und Netzauslastung zu achten,“ Agora Verkehrswende, Aachen, 2023.
- [6] I. S. Michael Sterner, Energiespeicher, Bedarf - Technologien - Integration, 2. Auflage, Berlin: Springer Vieweg, 2017.
- [7] Nissan, „Nissan Motor Corporation,“ Nissan Global, [Online]. Available: <https://www.nissan-global.com/EN/STORIES/RELEASES/nissan-works-to-power-v2x/>. [Zugriff am 30 November 2023].
- [8] ADAC, „ADAC,“ ADAC, [Online]. Available: <https://www.adac.de/rund-ums-fahrzeug/elektromobilitaet/info/bidirektionales-laden/>. [Zugriff am 30 November 2023].
- [9] Teslamag, „Teslamag,“ [Online]. Available: <https://teslamag.de/news/powerwall-vorbote-tesla-elektroautos-experte-bidirektional-61617>. [Zugriff am 30 November 2023].
- [10] M. M. A. O. u. T. K. Sebastian Faller, „Bidirektionales Laden: Von der Last zur Lösung!,“ [Online]. Available: chrome-extension://efaidnbnmnnibpcajpcglclefindmkaj/https://www.ffe.de/wp-content/uploads/2020/03/20200401_Bidirektionales-Laden_Von_der_Last_zur_Loesung.pdf. [Zugriff am 30 November 2023].

- [11] Honda, „Honda,“ 2019. [Online]. Available: https://global.honda/en/innovation/CES/2019/wireless_vehicle-to-grid.html. [Zugriff am 30 November 2023].
- [12] VattenFall, „Vattenfall InCharge Wissens-Hub,“ 15 Spetember 2023. [Online]. Available: <https://incharge.vattenfall.de/wissens-hub/articles/was-ist-smart-charging-und-wie-konnen-elektroautofahrer-davon-profitieren>. [Zugriff am 30 November 2023].
- [13] S. R. C. G. K. G. Norman Ihle, „DATENAUSTAUSCHFORMATE FÜR DEMAND RESPONSE – EVALUATION UND FALLBEISPIEL MIT OPENADR,“ in *Symposium Energieinnovation*, Graz - Österreich, 2016.
- [14] L. Röhl, „Simulative Auslegung solargespeister Elektroladestationen mit Pufferbatterie,“ Technische Hochschule Ingolstadt, Ingolstadt, 2023.
- [15] N. W. D. R. D. K. V. B. Leonie Schmidt, „NRW.Energie 4Climate,“ September 2023. [Online]. Available: <https://www.energy4climate.nrw/aktuelles/newsroom/elektroautos-als-mobiler-speicher-und-beschleuniger-der-energiewende-voraussetzungen-muessen-schnell-geschaffen-werden>. [Zugriff am 02 Dezember 2023].
- [16] M. D. Müller, „BDL - Bidirektionales Lademanagement,“ München, 2023.
- [17] S. Wanger, „IAB-FORSCHUNGSBERICHT,“ Nürnberg, 2020.
- [18] S. P. E. U. Dietmar Hobler, „Lage der Arbeitszeit von abhängig Beschäftigten 1996–2020,“ Hans-Böckler-Stiftung, 2021.
- [19] B. f. W. u. Klimaschutz, „Bundesministerium für Wirtschaft und Klimaschutz,“ [Online]. Available: https://www.bmwk.de/Navigation/DE/Themen/themen.html?cl2Categories_LeadKeyword=wirtschaftsbranchen. [Zugriff am 24 03 2024].
- [20] AG Energiebilanzen e.V., „AGEB,“ [Online]. Available: <https://ag-energiebilanzen.de/>. [Zugriff am 04 02 2024].
- [21] L. Fahrmeir, R. Künstler, I. Pigeot und G. Tutz, Statistik: Der Weg zur Datenanalyse, Berlin, Heidelberg: Springer Berlin Heidelberg, 2007.
- [22] G. P. M. u. B. A. C. Introduction to the Practice of Statistics" von David S. Moore, Introduction to the Practice of Statistics", WH Freeman, 2014.
- [23] B. Klein, „Numerisches Python,“ Carl Hanser Verlag, München, 2023.
- [24] D.-I. C. Lange, „Fraunhofer IISB,“ Fraunhofer-Institut für Integrierte Systeme und Bauelementetechnologie, [Online]. Available: <https://www.energy->

seeds.org/de/schwerpunkte/lastverschiebung_sekundaerenergiespeicher.html. [Zugriff am 24 03 2024].

- [25] P.-D.-I. T. Schneiders, „Grundlagen und Kennzahlen der Energiespeicherung,“ TH-Köln, 2021.
- [26] S. B. Amt, „Destatic,“ [Online]. Available: <https://www-genesis.destatis.de/genesis/online?operation=themes&code=42#abreadcrumb>. [Zugriff am 03 02 2024].
- [27] Bundesministerium für Wirtschaft und Klimaschutz, Herr Dr. Kopp, „Bundesministerium für Wirtschaft und Klimaschutz,“ 16 12 2021. [Online]. Available: <https://www.bmwk.de/Redaktion/DE/Artikel/Branchenfokus/Industrie/branchenfokus-chemie-pharmazie.html>. [Zugriff am 27 03 2024].
- [28] Bundesministerium für Wirtschaft und Klimaschutz, „Bundesministerium für Wirtschaft und Klimaschutz,“ [Online]. Available: <https://www.bmwk.de/Redaktion/DE/Textsammlungen/Branchenfokus/Industrie/branchenfokus-automobilindustrie.html#:~:text=Zur%20deutschen%20Automobilindustrie%20z%C3%A4hlen%20die,23.000%20weniger%20als%20im%20Vorjahr..> [Zugriff am 24 03 2024].
- [29] Bundesministerium für Umwelt, Naturschutz und nukleare Sicherheit (BMU), „Bundesministerium für Umwelt, Naturschutz und nukleare Sicherheit (BMU),“ 1. November 2016. [Online]. Available: <https://www.bmu.de/publikation/klimaschutzplan-2050>.
- [30] Alois Freidhof, Bundesamt für Energie, „Batterien für Elektrofahrzeuge,“ Bundesamt für Energie BFE, Bern/Ittingen, 2023 April.
- [31] M. R. P. M. M. N. A. H. Tobias Fleiter, „Langfristszenarien 3 – Hauptszenarien Modul Industrie,“ Bundesministerium für Wirtschaft und Energie (BMWi), 2021.

Anhang

Simulationdaten:

- step_1_interpoliere_lastprofil.py
- step_2_berechne_tagesmittel_und_tageshoch.py
- step_3_berechne_akku_verfuegbarkeiten.py
- step_4_berechne_batterie_zielzustaende.py
- step_5_berechne_daily_max.py
- step_6a_calculate_szenario_simple.py
- step_6b_calculate_szenario_ladebremse.py
- step_6c_calculate_szenario_end-of-day.py
- step_6d_calculate_szenario_bidi.py
- step_x_erstelle_grafiken.py
- szenario_helpers.py
- Ergebnisse aller Berechnung im Ordner results (CSV-Format)
- Zwischenergebnisse aller Berechnung im Ordner intermediates (CSV-Format)
- Lastprofile und Beschäftigungsverteilung im Ordner inputs (CSV-Format, Exceldatei)
- Grafiken im den Ordnern Plots I_0, I_1, I_5, I_6, I_13 (PNG-Format)