



TECHNISCHE
UNIVERSITÄT
WIEN
Vienna University of Technology



DIPLOMARBEIT

Simulation von Demand Side Management mit frequenzabhängigen Lastprofilen in Inselnetzen

Ausgeführt zum Zwecke der Erlangung des akademischen Grades

Diplom-Ingenieur

unter der Leitung von

O.Univ.Prof. Dipl.-Ing. Dr.-Ing. Günther Brauner

Institut für Elektrische Anlagen und Energiewirtschaft (E373)
Fakultät für Elektrotechnik und Informationstechnik
Technische Universität Wien

betreut von

Univ.Ass. Dipl.-Ing. Alfred Einfalt

Institut für Elektrische Anlagen und Energiewirtschaft (E373)
Fakultät für Elektrotechnik und Informationstechnik
Technische Universität Wien

eingereicht an der

Technischen Universität Wien

Fakultät für Elektrotechnik und Informationstechnik

von

Franz Zeilinger, BSc

Matrikelnummer 0325791
Quellenweg 1, 4643 Pettenbach

Wien, Dezember 2010

Danksagung

Eine Abschlussarbeit ist eine besondere Leistung, die erbracht werden muss, um ein naturwissenschaftliches Studium abzuschließen. Niemals wäre es mir gelungen, diese Arbeit zu vervollständigen, wenn ich nicht von vielen Seiten dementsprechende Unterstützung erhalten hätte.

Deshalb möchte ich mich bei Herrn O.Univ.Prof. Dipl.-Ing. Dr.-Ing. Günther Brauner bedanken, der mein Interesse für das Thema dieser Arbeit geweckt hat. Erst aus den Gesprächen und Diskussionen mit ihm hat sich diese Diplomarbeit entwickelt. Univ.Ass. Dipl.-Ing. Alfred Einfalt danke ich besonders für seine praktische Unterstützung und Ratschläge.

Generell möchte ich allen Lehrenden an der Technischen Universität Wien insbesondere jene der Fakultät für Elektrotechnik und Informationstechnik danken, dass sie mir das weite Feld der Elektrotechnik und der Energietechnik nahegebracht haben. Das Studium gestaltete sich durch ihr Bemühen zu einer interessante Reise, die ich jederzeit wieder antreten würde.

Dank gilt auch meinen Studienkollegen und Freunden, die ebenfalls dazu beitrugen, dass meine Studienzeit zu einem ganz besonderen Lebensabschnitt wurde.

Ganz besonders möchte ich meiner Freundin, Elke Ebner, danken, die mir durch ihre seelische und moralische Unterstützung das Studium sowie das Erstellen dieser Arbeit erleichtert hat. Auch danke ich ihr für die Geduld, die sie aufgebracht hat und dies nicht nur beim Korrekturlesen.

Meinen Eltern Franz und Maria Anna Zeilinger danke ich dafür, dass sie maßgeblich dazu beigetragen haben, dass ich zu dem geworden bin, der ich nun bin. Sie und meine Brüder Herbert und Gerhard haben mir ein Zuhause bereitet, ohne dem ich das Studium niemals hätte abschließen können.

Zusammenfassung

Um jene Fragen klären zu können, die sich bei einer Realisierung von Demand Side Management (DSM) speziell im Haushaltsbereich stellen, wurde im Rahmen dieser Arbeit ein Simulationsmodell entwickelt, das einerseits Verbraucher im Haushalt, andererseits die Auswirkungen von verschiedenen DSM-Systemen auf deren Leistungsaufnahme nachbilden kann.

Dazu wurden die Verbrauchergruppen in einem Haushalt in ihrem Verhalten nachgebildet. Die einzelnen Verbrauchergruppen werden mit Hilfe von so genannten Geräteklassen modelliert. Dabei wurde darauf geachtet, dass das simulierte Verhalten dieser Geräteklassen möglichst ähnlich der von realen Geräten ist.

Die zu untersuchenden DSM-Systeme sollen einen einfachen Aufbau aufweisen und ohne zusätzliche Kommunikation zwischen den Verbrauchern auskommen. Die Information über die Notwendigkeit eines Eingriffs in die Leistungsaufnahme der Verbraucher soll aus der Netzfrequenz ermittelt werden. Dazu wurden verschiedene Algorithmen entwickelt, mit denen ein DSM-System ausgestattet werden kann. Diese Algorithmen lassen sich zu verschiedenen DSM-Systemen kombinieren um die Auswirkungen von diesen untersuchen zu können.

Mit Hilfe von bereits bekannten Lastgängen, durchgeführten Messungen und weiteren Daten konnte ein Modell der Verbraucher für einen gemittelten Tageslastgang erstellt und überprüft werden.

Dieses Modell diente in weiterer Folge als Grundlage für erste Untersuchungen von DSM-Systemen. Mit Hilfe dieser Ergebnisse können bereits erste Anforderungen an eine Realisierung von DSM gewonnen werden.

Abstract

In order to answer the questions that arise especially in the household sector with an implementation of Demand Side Management (DSM), a simulation model was developed in this thesis, which can reproduce the behavior of household devices on the one hand and the effects of various DSM-systems on their power consumption on the other hand.

For this purpose, the different groups of consumers in a household were simulated in their behavior. Each group of consumers is modeled by using so called device-classes. Care was taken that the simulated behavior of these classes is as similar as possible to that of real devices.

The DSM-systems to be examined should not require any additional communication between the consumers and have a simple structure. The information on the need for an intervention in the power consumption of the devices is to be determined from the line frequency. Thus, different algorithms a DSM-system can be equipped with have been developed. These algorithms can be combined to form different DSM-systems to examine the effects they may have.

With the help of already known load curves, through measurements and additional data, a model of the devices of a household for an average daily load was established and verified.

This model was used subsequently as the basis for initial investigations of different DSM-systems. Using these results, initial demands for a realization of DSM can already be obtained.

Inhaltsverzeichnis

1. Einleitung	1
1.1. Motivation für DSM	1
1.2. Bisherige Realisierungen	2
1.2.1. IRON – Integral Resource Optimisation Network	2
1.2.2. GridFriendly™ Appliance Controller	2
1.2.3. Projekt DINAR – BEMI.....	3
1.2.4. digitalSTROM®	4
1.3. Ansatz für eine Simulation	4
2. Simulationssoftware.....	7
2.1. Parameter.....	7
2.1.1. Simulationseinstellungen.....	9
2.1.2. Geräteeigenschaften.....	10
2.1.3. DSM-Funktion	11
2.2. Simulation der Verbraucher.....	11
2.2.1. Device.....	12
2.2.2. Cont_Operation	13
2.2.3. Thermal_Storage	13
2.2.3.1. Model eines thermischen Speichers	13
2.2.3.2. Simulation des thermischen Speichers	15
2.2.3.3. Parameter und Auswirkung auf das Simulationsergebnis.....	16
2.2.3.4. Abweichungen Modell – Realität	19
2.2.4. Scheduled_Operation	19
2.2.4.1. Einsatzplan	20
2.2.4.2. Ermittlung der aufgenommenen Leistung.....	21
2.2.4.1. Stand-by-Verbrauch	22
2.2.5. Periodic_Operation	23
2.2.5.1. Erstellung des Einsatzplanes.....	23
2.2.5.2. Beispiel einer Instanz von Periodic_Operation	24
2.2.6. Probable_Operation	24
2.2.6.1. Parameter.....	24
2.2.6.2. Erstellung des Einsatzplanes.....	25
2.2.6.3. Auswirkungen der Parameter auf das Simulationsergebnis.....	25
2.2.7. Loadcurve_Operation.....	27
2.2.7.1. Parameter.....	27
2.2.7.2. Erstellung des Einsatzplanes.....	29
2.3. Simulation der DSM-Funktionalität.....	31
2.3.1. DSM-Input.....	31

2.3.1.1.	No_DSM_Function.....	33
2.3.1.2.	Frequency_Response_Simple	33
2.3.1.3.	Frequency_Response_Priority.....	37
2.3.2.	DSM-Output.....	38
2.3.2.1.	Turn_Off.....	39
2.3.2.2.	Turn_Off_Stand_by.....	42
2.3.2.3.	Reduce_Input_Power	42
2.3.2.4.	Change_Temp_Set	43
2.3.2.5.	Postpone_Start.....	45
2.3.2.6.	Pause_Programm	48
2.4.	Frequenzdaten	50
2.5.	Simulationsdurchführung und Datenausgabe.....	50
2.5.1.	Erzeugung des Gerätekollektivs.....	50
2.5.2.	Simulationsablauf	51
2.5.3.	Einzelsimulation	52
2.5.4.	Simulationsreihe	52
2.5.5.	Nachbearbeitung der Daten	53
2.5.6.	Speichern der Daten	53
2.5.7.	Anzeigen der Daten.....	54
2.6.	Programmoberfläche - GUI.....	54
2.6.1.	Hauptfenster.....	55
2.6.1.1.	Menü Datei.....	56
2.6.1.2.	Menü Parameter	56
2.6.1.3.	Menü Frequenzdaten.....	56
2.6.1.4.	Menü Simulationsreihe	57
2.6.2.	Einstellungen für Simulationsreihen.....	57
3.	Ermittlung der Parameter für das Simulationsmodell.....	59
3.1.	Vergleichsdaten	59
3.2.	Bürogeräte	60
3.2.1.	Durchschnittlicher Stand-by-Verbrauch.....	61
3.2.2.	Durchschnittliche Anschlussleistung.....	61
3.2.3.	Zusammenfassung der Parameterwerte.....	62
3.3.	Audio-Video-Geräte.....	63
3.3.1.	Durchschnittlicher Stand-by-Verbrauch.....	63
3.3.2.	Durchschnittliche Anschlussleistung.....	64
3.3.3.	Zusammenfassung der Parameterwerte.....	64
3.4.	Beleuchtung.....	66
3.5.	Kühlgeräte	67
3.5.1.	Ermittlung der thermischen Parameter	67
3.5.2.	Kühlschränke.....	68

3.5.3. Gefriergeräte	72
3.6. Waschmaschinen	76
3.6.1. Lastkurven.....	76
3.6.2. Zusammenfassung der Parameterwerte	77
3.7. Geschirrspüler.....	80
3.7.1. Lastkurve.....	80
3.7.2. Zusammenfassung der Parameterwerte	80
3.8. Gesamtverbrauch.....	82
4. Potentiale der verwendeten Gerätegruppen.....	85
4.1. Bürogeräte.....	85
4.2. Audio-Video-Geräte.....	85
4.3. Beleuchtung.....	86
4.4. Kühlschränke und Gefriergeräte	86
4.5. Waschmaschinen und Geschirrspüler	86
5. Simulation von Verbrauchern mit DSM	87
5.1. Verschiebung Lastspitzen.....	87
5.1.1. Einfluss der Standardabweichung der Frequenzschwelle	88
5.1.2. Rollout von DSM.....	90
5.1.3. Verhalten von Waschmaschinen	90
5.1.4. Verhalten von Geschirrspülern	93
5.1.5. Verhalten von Kühlgeräten.....	95
5.1.6. Einsatz identer Chips in allen Geräten.....	96
5.1.7. Zusammenfassung	98
5.2. Reaktion auf kurzen Frequenzeinbruch	101
5.3. Reaktion auf langen Frequenzeinbruch.....	103
5.4. Reaktion auf Frequenzverlauf bei einer Großstörung.....	105
6. Schlussfolgerungen	111
6.1. Erweiterungen der Software und des Simulationsmodells	111
6.2. Weiterführende Untersuchungen.....	112
Anhang.....	115
A Literaturverzeichnis.....	115
B Abbildungsverzeichnis.....	117
B1 Abbildungen.....	117
B2 Diagramme	117
C Tabellenverzeichnis	121

1. Einleitung

Angesichts des Klimawandels, der laut internationalen Gremien im Zusammenhang mit dem weltweiten Verbrauch von fossilen Energieträgern steht, befindet sich derzeit die Energiebranche in einem Umbruch [1: S.2].

Maßnahmen zur Förderung von erneuerbaren Energieträgern haben in Europa zu einer massiven Weiterentwicklung der regenerativen Energieerzeugung geführt. Für die regenerative Stromerzeugung werden derzeit hauptsächlich Kleinwasserkraftwerke, Biomasse-, Photovoltaik- und Windenergieanlagen in Betracht gezogen [2: S.1].

Insbesondere Windenergie- und Photovoltaik-Anlagen sind angesichts der hohen Volatilität des Dargebots sehr schwer zu prognostizieren [3: S.6]. Laufend werden neue Projekte gerade im Bereich der Windenergie umgesetzt, ohne dass, aufgrund von langwierigen Genehmigungsverfahren, das Übertragungsnetz und die Kapazität an Regelkraftwerken mit dieser Entwicklung schritthalten [2: S.1]. Neben dieser Belastung sind die Übertragungsnetze in Europa aufgrund von neuen Marktmechanismen zu einer Plattform für die Übertragung von immer höher werdenden Energieströmen geworden [4: S.5]. Gerade durch das Auftreten von großen Netzstörungen zeigt sich, dass der Spielraum im Übertragungsnetz immer kleiner wird [5].

Um einerseits die stark schwankende Stromerzeugung aus erneuerbaren Energieträgern auszugleichen und damit die Übertragungsnetze zu entlasten oder andererseits den Übergang zu dezentralen Erzeugungsstrukturen zu ermöglichen, die sogar kleine, autonome Energiesysteme erlauben, muss in Zukunft neben der Regelung der Erzeugung von Energie auch der Verbrauch dieser beeinflussbar werden.

Die Möglichkeiten von lastseitigen Steuerungseingriffen durch sogenanntes *Demand Side Management* (DSM) speziell im Haushaltsbereich werden daher in dieser Arbeit näher behandelt.

1.1. Motivation für DSM

Die Bilanz zwischen Erzeugung und Verbrauch von elektrischer Energie muss immer ausgeglichen sein. Möglichkeiten um für dieses Gleichgewicht zu sorgen sind das Speichern bzw. das Übertragen von überschüssiger Energie. Diese Möglichkeiten sind jedoch immer mit Verlusten verbunden, abgesehen von den eingangs erwähnten Problemen, die sich durch eine weiträumige Übertragung auf bereits stark belasteten Übertragungsnetzen ergeben.

Demand Side Management (DSM) soll eine andere Möglichkeit bieten, ein Gleichgewicht aus Energieerzeugung und Energieverbrauch herzustellen, indem der Verbrauch von Energie beeinflusst wird und z.B. diesen in Zeiten verschiebt, in denen die Einspeisung durch fluktuierende Erzeuger hoch ist [6: S.4].

Untersuchungen zeigen, dass gerade im Haushalts- und Kleingewerbebereich für DSM ein beachtliches Potential zur Verfügung steht, dieses jedoch, mit Ausnahme von Rundsteuersignalen, bisher nicht erschlossen wird [1, 6, 7].

Um dieses Potential nutzen zu können, ist insbesondere bei der Umsetzung von DSM im Haushalt die Frage nach den damit verbundenen Kosten zu klären. Ist das DSM-System nur mit hohem technischen und finanziellen Aufwand realisierbar, können nur wenige Verbraucher damit gesteuert werden, da sich die Einbindung von Verbrauchern mit geringer Anschlussleistung nicht rentiert [2: S.VI]. Könnte ein DSM-System sehr günstig implementiert werden, z.B. durch hohe Stückzahlen und einfachen Aufbau, kann man davon ausgehen, dass auch mehr Verbraucher für eine Leistungsbeeinflussung zur Verfügung stehen werden.

Neben diesen Fragen muss auch noch die Bereitschaft des Konsumenten beleuchtet werden, dass „seine“ Geräte sich am Ausgleich der Netzbilanz beteiligen. Bleibt diese vom Konsumenten unbemerkt, kann man davon ausgehen, dass in diesem Fall der Einsatz von DSM ohne weiteres ermöglicht wird.

Kommt es jedoch zur bemerkbaren Beeinflussung des Geräte- und damit des Nutzungsverhaltens, muss eine entsprechende Vergütung vorhanden sein, damit diese Einschränkungen akzeptiert werden. Auch könnten Anreize durch den Gesetzgeber oder die spezielle Netzsituation dazu führen, dass der Konsu-

ment eine höhere Akzeptanz von DSM aufbringt bzw. aufbringen muss. Insbesondere für eine Realisierung von weitgehend energieautonomen Inselnetzen mit hauptsächlich regenerativer Energieerzeugung ist ein Einsatz von DSM von großer Bedeutung.

Weitere Fragen, die sich beim Einsatz von DSM stellen, sind beispielsweise:

- Auswirkungen bei einem Einsatz in vielen Geräten
- Können die erwarteten Potentiale auch genutzt werden?
- Wie stark wird das Nutzerverhalten beeinträchtigt?

Um sich diesen und weiteren Fragen zu nähern und sie eventuell beantworten zu können, müssen zuerst konkretere Angaben zum DSM-System selbst gemacht werden. Dazu sollen hier zunächst bereits vorhandene DSM-Systeme vorgestellt werden.

1.2. Bisherige Realisierungen

DSM ist schon seit längerer Zeit ein Thema von diversen Projekten, die teilweise schon realisiert wurden. Im Folgenden soll kurz ein Überblick über jene Forschungen gegeben werden, die im Zuge der Recherche für diese Arbeit genauer betrachtet wurden. Diese Liste erhebt aber keinerlei Anspruch auf Vollständigkeit.

1.2.1. IRON – Integral Resource Optimisation Network

IRON (*Integral Resource Optimisation Network*) ist ein Forschungsprojekt des Instituts für Computertechnik der TU Wien. Es untersucht die Umsetzbarkeit, Anwendbarkeit, die notwendigen Algorithmen sowie deren technischen Implementierung in dem interdisziplinären Feld von „Distributed DSM“. Die zentrale Idee dieses Projektes ist die Zurverfügungstellung einer modernen Informations- und Kommunikationsinfrastruktur für zukünftige effiziente Energiesysteme [8: S.19].



Abbildung 1.1: Design der DSM-Interface-Einheit IRON Box (aus [9: S.104])

Als erste Umsetzung wurde ein System zur lastseitigen Primärregelung entwickelt und implementiert, das Verbraucher einbindet, bei denen träge thermische Prozesse ablaufen. Der in [9] vorgestellte Algorithmus soll robust gegen Ausfälle der Kommunikationsinfrastruktur sein, da in erster Linie auf Schwankungen der Netzfrequenz reagiert wird und so die notwendige Kommunikation zwischen den Lastknoten keine Echtzeitbedingung erfüllen muss.

1.2.2. GridFriendly™ Appliance Controller

Ein Ansatz für DSM, der sich besonders mit von der Netzfrequenz abhängiger Steuerung beschäftigt, ist das Teilprojekt „GridFriendly™ Appliance Controller“ (GFA Controller) der „GridWise™“-Initiative in den USA. Dieser Controller (siehe Abbildung 1.2) soll in möglichst viele Geräte integriert werden. Fällt die Netzfrequenz unter einen bestimmten Wert, reagiert der Controller mit einer Reduzierung der Leistung des angesteuerten Gerätes. Dabei werden sonst keine weiteren Kommunikationskanäle benötigt. Über zufällig variierte Verzögerungszeiten soll verhindert werden, dass alle Controller ihre Last gleichzeitig abwerfen.

Ziel dieses Controllers ist es, dass die Verbraucher bei einer einbrechenden Netzfrequenz einen möglicherweise bevorstehenden Black-Out vermeiden helfen, sich daher „netzfrendlich = gridfriendly“ verhalten [9: S.24–25, 10].

Bisher wurden 50 elektrische Warmwasserboiler und 150 Wäschetrockner in Privathaushalten mit dem Controller ausgestattet. Bei diesem Versuch wurden drei der Controller im Raum von Washington und Oregon ein Jahr lang überwacht. Im Schnitt mussten die Controller einmal pro Tag eine Leistungsreduktion des Gerätes vornehmen. Eine Umfrage unter den Bewohnern der betroffenen Häuser zeigte, dass die Leistungsbeeinflussung von ihnen kaum bemerkt wurde bzw. nur wenige Unannehmlichkeiten verursachten [10: S.5].

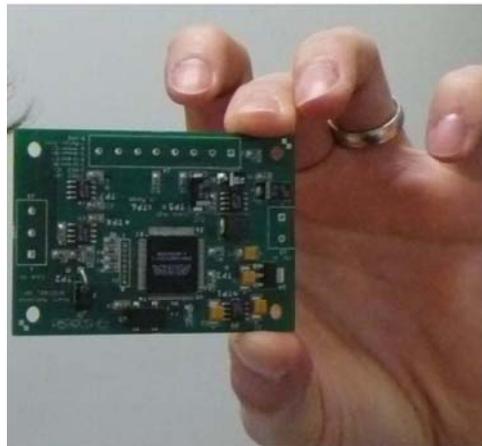


Abbildung 1.2: Der „GridFriendly™ Appliance Controller“ (aus [9: S.25])

1.2.3. Projekt DINAR – BEMI

Das Forschungsprojekt DINAR hatte sich das Ziel gesetzt, eine technische und wirtschaftliche Lösung für ein bidirektionales Energiemanagement im Niederspannungsnetz zu finden, wobei Verbrauch und Erzeugung als Einheit betrachtet werden.

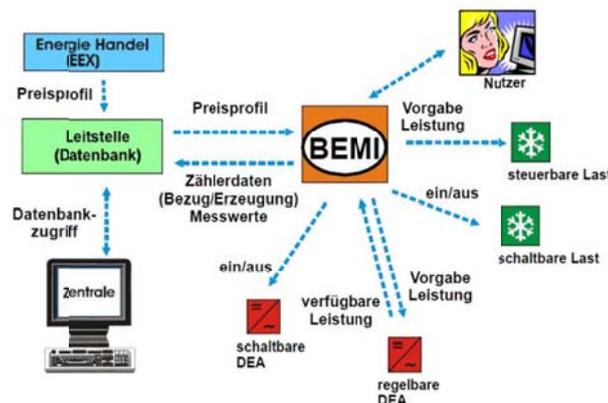


Abbildung 1.3: Kommunikationsstruktur des BEMI, ©ISET (aus [6: S.14])

Allerdings geht man strategisch einen neuen Weg, der sich wesentlich von bisherigen Lösungen unterscheidet. Die zentrale Steuerung (wie z.B. beim virtuellen Kraftwerk) wird durch eine dezentrale ersetzt bei gleichzeitigem Wegfall der Notwendigkeit einer Online-Kommunikation. An Bedeutung gewinnt der dezentrale Netzanschlusspunkt, der messtechnisch „beobachtet“ wird.

An dieser technischen und juristischen Schnittstelle wird das sogenannte „*bidirektionale Energiemanagement Interface* – BEMI“ angesiedelt und entwickelt, welches Energieerzeuger und Verbraucher gleichberechtigt managen soll und den Netzanschlusspunkt unter qualitätssichernden Kriterien beobachtet und bewertet. Seit November 2005 werden erste Implementierungen des BEMI getestet und laufend erweitert und optimiert [6].

1.2.4. digitalSTROM®

digitalSTROM.org ist eine Non-Profit-Organisation, die am 7.7.2007 an der eidgenössischen Technischen Hochschule (ETH) Zürich gegründet wurde, um digitalSTROM® zu einem weltweiten Standard für intelligente Heimnetze zu entwickeln.

Herzstück des Systems ist der sogenannte „dSID-Chip“ (siehe Abbildung 1.4). Auf diesem Hochvoltchip befinden sich Prozessor, Speicher, Netzteil und Kommunikationsmodul. Der Chip kann in jedes Gerät eingebaut werden und soll z.B. auch in Form einer Lusterklemme für eine nachträgliche Nachrüstung erhältlich sein.

Neben einer Protokollierung des Verbrauchs jedes mit einem „dSID-Chips“ ausgestatteten Geräts können diese Geräte auch zentral angesteuert werden. Die Kommunikation zwischen den Stromchips erfolgt über die Hausinstallation in Form einer Datenübertragung im Basisband. Daher sind die Chips nicht außerhalb ihres Stromkreises sichtbar.

Die Projektverantwortlichen gehen davon aus, dass in Zukunft der Strompreis von der Tageszeit abhängig sein wird. Mit Hilfe von digitalSTROM® sollen dann alle Geräte so gesteuert werden, dass sie möglichst kostensparend betrieben werden können [1: S.5–6].

Im Jänner 2011 sollen erste Komponenten über die Vertriebskanäle der Herstellerfirmen erhältlich sein [11].

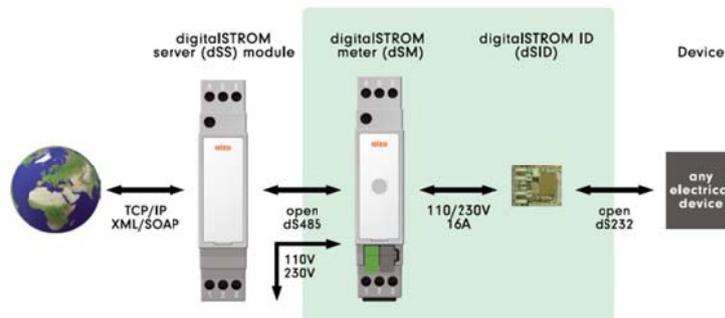


Abbildung 1.4: Aufbau des digitalSTROM® Systems (aus [12])

1.3. Ansatz für eine Simulation

Zu Beginn dieser Arbeit stand der Wunsch, ein DSM-System zu entwickeln, das möglichst einfach in seinem Aufbau sein sollte. Die meisten Entwicklungen gehen jedoch, wie in den vorherigen Abschnitten erläutert, hin zu Systemen, die eine mehr oder minder aufwendige Kommunikation zwischen den Geräten erfordern.

Das zu entwickelnde System sollte allerdings nur selbstständig den Netzzustand mit Hilfe der Netzfrequenz ermitteln und darauf dementsprechend reagieren, ähnlich dem GFA Controller (siehe Abschnitt 1.2.2).

Welche Potentiale im Haushaltsbereich unter Einbeziehung aller elektrischen Energieverbraucher vorhanden sind, wurde bereits in [1] bzw. [7] genauer untersucht. Bei der Entwicklung des DSM-Systems sollte nun versucht werden, möglichst viel von diesem Potential im Haushalt auch nutzen zu können.

Sofort stellten sich, neben den bereits eingangs angeführten Fragen, weitere, die vor einer konkreten Realisierung eines Systems geklärt werden müssen:

- Welche Funktionen muss eine DSM-Einheit zu Verfügung stellen? Lassen sich diese mit Hilfe einfacher, am besten standardisierter Bauelemente realisieren?
- Welche Anforderungen müssen an die Geräte gestellt werden um sie in das DSM-System einbinden zu können?
- Wie stark muss eine DSM-Einheit in das Gerät integriert werden?
- Welche Anforderungen müssen an die Hardware der DSM-Einheit gestellt werden (Messgenauigkeit, Schaltvermögen, ...)

- Wie kann erreicht werden, dass das System, trotz fehlender Kommunikation zwischen den Geräten, nicht zu ungünstigen Netzzuständen führt?
- Kann das System auch dazu verwendet werden, kleine autonome Inselnetze zu unterstützen bzw. zu ermöglichen?

Der Zugang zur Beantwortung dieser Fragen führte zu der Notwendigkeit einer vorherigen Simulation von möglichen Realisierungen von verschiedenen DSM-Systemen.

Neben der Modellierung des DSM-Systems muss auch das Verhalten der Verbraucher eines Haushaltes nachgebildet werden, um verwertbare Aussagen über das DSM-System erhalten zu können. Für die Modellierung von Verbrauchern gibt es zwar bereits umgesetzte Modelle (z.B. in [13]), diese können aber nicht dazu benutzt werden, darin ein DSM-System in der erforderlichen Feinheit einzubinden.

Aus diesem Grund wurde eine eigene Möglichkeit der Simulation von Verbrauchern entwickelt, die dann in weiterer Folge mit einem DSM-System in ihrer Leistungsaufnahme beeinflusst werden können. In Kapitel 2 wird die zu diesem Zweck entwickelte Software vorgestellt.

Mit Daten aus [1] wurde darauf aufbauend ein Modell für die Verbraucher in einem Haushalt ermittelt. Dieses wird in Kapitel 3 genauer erläutert.

Mit Hilfe dieses Modells konnten erste DSM-Systeme in ihren Eigenschaften untersucht werden, um die vorhin erwähnten Fragen zu klären. Erste Ergebnisse dazu werden in Kapitel 5 präsentiert.

Die in dieser Arbeit entwickelte Modelle und Methoden sind dazu vorgesehen, erweitert und verbessert zu werden und sollen dann in weiterer Folge helfen, genauere Aussagen über den Einsatz von DSM in Haushalten zu ermöglichen.

Mit diesen Erkenntnissen soll die reale Umsetzung eines DSM-Systems unterstützt werden.

2. Simulationssoftware

Im Zuge dieser Arbeit wurde eine Simulationssoftware entwickelt, mit der es ermöglicht wird, die Auswirkungen von verschiedenen DSM-Systemen zu untersuchen. Der Programmwurf erfolgte mit dem Ziel, möglichst flexibel in der Definition von Verbrauchergruppen und DSM-Funktionen inklusive deren Kombination zu sein, sowie eine künftige Erweiterung aller Funktionalitäten zu ermöglichen.

Die Software wurde mit Hilfe von MATLAB geschrieben und verwendet einige der darin implementierten Funktionen. Um im folgenden Abschnitt Verweise auf Programmteile (Klassennamen, Variablen, Parameternamen, Funktionen, ...) hervorzuheben, werden diese in dieser Schreibweise im Text hervorgehoben.

Bei der Beschreibung des Programms wird nur soweit auf die tatsächlich ausgeführte Programmierung eingegangen, wie es für das Verständnis des Verhaltens und des Zustandekommens der Ergebnisse notwendig ist. Vordergründig soll in diesem Kapitel beschrieben werden, wie sich die verschiedenen Parameter und Methoden auf die Simulation auswirken. Für Erklärungen zu den programmatischen Realisierungen der einzelnen Funktionen sei an dieser Stelle auf die Kommentare im Quellcode der Software verwiesen.

2.1. Parameter

Für einen Simulationsdurchlauf mit vielen verschiedenen Gerätetypen müssen viele unterschiedliche Parameter für die jeweiligen Geräte sowie deren DSM-Funktion definiert und dem Programm übergeben werden. Um das möglichst einfach zu gestalten, soll die Parametereingabe auf eine intuitive und unkomplizierte Weise erfolgen.

Die wichtigsten Einstellungen für die Simulation lassen sich direkt im GUI des Programms ändern (siehe Abschnitt 2.5).

Für die Geräte- und DSM-Parameter wäre eine Eingabe über ein GUI sehr umständlich bzw. unübersichtlich, da sehr viele Parameter jeweils unterschiedlich für verschiedene Gerätetypen definiert werden müssen. Daher erfolgt die Parametereingabe über eine EXCEL-Tabelle, in der die verschiedenen Parameter definiert werden. Damit können sehr einfach viele unterschiedliche Parametersets definiert, gespeichert und verwaltet werden. Weiters können innerhalb der EXCEL-Tabelle Formeln und Zellverweise verwendet werden, um so viele Parameterwerte über ein zentrales Eingabefeld zu verändern. Das Simulationsprogramm liest die Parameterdatei vor jedem Simulationsdurchlauf ein und wandelt die darin enthaltenen Daten entweder in eine Parameterliste für jede Geräteklasse oder direkt in Variablenwerte um.

Das Layout für die zu definierenden Daten ist sehr frei. Im Prinzip muss in einer EXCEL-Arbeitsmappe (Endung .xls) eine Tabelle mit dem Namen „Parameters“ definiert werden. Diese dient dazu, schnell jede andere EXCEL-Arbeitsmappe von einer Parameterdefinitionsdatei zu unterscheiden.

Tabelle 2.1 gibt ein Beispiel für den prinzipiellen Aufbau einer Parametertabelle in einer Parameterdatei. Diese kann vor der Simulation mit Hilfe des GUI festgelegt werden. Weiters bietet das GUI die Möglichkeit, schnell Zugriff auf die aktuelle Parameterdatei zu erhalten. Näheres dazu findet sich im Abschnitt 2.5.

Mit Hilfe von bestimmten, vorab im Programm definierten Bezeichnungen, wie z.B. den Gerätetypen, kann das Programm die einzelnen Parametergruppen in der Tabelle unterscheiden. Dabei wird die Tabelle anhand der bekannten Bezeichnungen in Bereiche eingeteilt. Ein Bereich geht von einer Bezeichnung bis exklusive der nächsten. Die Teilbereiche werden dann separat wiederum anhand von erkannten Bezeichnungen geteilt, solange bis keine weitere Teilung mehr möglich ist. Es gibt zwei Hauptbereiche:

Im ersten Bereich können, alternativ zum GUI, die Simulationseinstellungen definiert werden (siehe nächster Abschnitt). Dieser Bereich wird dadurch gekennzeichnet, dass in der ersten Zeile dieses Bereiches die Bezeichnung „Simulation Settings:“ steht. In den darauf folgenden Zeilen werden dann die Simulationseinstellungen angeführt.

Der zweite Hauptbereich wird durch die Bezeichnung „Device Settings:“ eingeleitet. Hier werden die einzelnen Geräte und deren Parameter definiert. Pro Gerät lassen sich wieder zwei Bereiche unterscheiden.

Der Bereich unterhalb des Gerätenamens enthält die Parameter für die Erzeugung der Geräteinstanzen (siehe Abschnitt 2.1.2). Danach muss die Bezeichnung „DSM-Settings:“ folgen, die den Bereich kennzeichnet, in dem die Parameter für die DSM-Funktion enthalten sind (siehe Abschnitt 2.1.3).

Die einzelnen Parameternamen müssen mit den im Programm definierten Variablennamen übereinstimmen, sonst kann keine Zuordnung erfolgen. Die Reihenfolge der Parameter in einem Bereich kann willkürlich festgelegt werden. Je nach Parameter werden unterschiedliche Bereiche berücksichtigt: Bei Date_Start z.B. wird nur eine Zeile, bei Device_Assembly der gesamte Bereich bis zur nächsten Bezeichnung übernommen, die erkannt wird. Im Programm werden dann die einzelnen Werte den richtigen Variablen zugeordnet.

Simulation Settings:									
Date_Start	19.02.2010 00:00								
Date_End	20.02.2010 00:00								
Sim_Resolution	hou	Nach mindestens einer leeren Zelle nach dem letzten Wert bzw. Einheit							
Number_User	50	kann hier ein beliebiger Inhalt stehen, er wird beim Einlesen der							
Use_DSM	1	Parameter ignoriert							
Use_Same_DSM	1								
Device_Assembly									
	Gerät_1	1							
	Gerät_2	0							
Device Settings:									
Gerätetyp_1									
	Parameter_1	100	W	20	%				
	Parameter_2	-1							
	Parameter_Liste_1	09:39	12:09	16:56	18:26	Uhr			
	'-> Std.Dev.:	180	180	120	60	min			
	Parameter_Liste_2	10	20	30	40	50	60	%	
DSM - Settings:									
DSM_Input_Mode									
	+--->	DSM_Input_Mode_1		-->	0	%			
	+--->	DSM_Input_Mode_2		-->	100	%			
	'--->	DSM_Input_Mode_3		-->	0	%			
	DSM_Option_1	1							
	DSM_Paramter_1	1	Hz	Uniform_Distr					
	DSM_Paramter_Liste_1	1	2	1,5	Hz				
	'-> Std.Dev.:	0	%						
DSM_Output_Mode									
	+--->	DSM_Output_Mode_1		-->	0	%			
	'--->	DSM_Output_Mode_2		-->	100	%			
	DSM_Paramter_2	1,5	°C	0	%				
Gerätetyp_2									
	Geräteparameter für Gerätetyp_2								
DSM - Settings:									
	DSM-Parameter für Gerätetyp_2								

Tabelle 2.1: Beispiel für den Aufbau einer Parametertabelle „Parameters“. Die farbigen Markierungen dienen ausschließlich der besseren Übersicht

Wo genau die Parameternamen in der Tabelle stehen, ist nicht wichtig. Es können beliebig viele Leerzeilen oder Spalten eingefügt werden. So können einfach Kommentare zu bestimmten Werten in der Tabelle angegeben werden. Einzige Ausnahme: Die Parameter müssen immer mindestens eine Spalte weiter rechts als die ihnen übergeordnete Bezeichnung stehen (siehe Beispiel in Tabelle 2.1, „Device Settings:“, „Gerätetyp_1“ und „Parameter_1“), damit eine einwandfreie Zuordnung erfolgen kann. Zur Abgrenzung der Parameterbereiche von Kommentaren reicht eine Leerzelle bzw. Leerzeile.

Hinter einem Parameternamen folgt der Wert, optional die Einheit und eventuell die Standardabweichung mit ihrer Einheit (siehe z.B. Parameter_1) bzw. die Angabe, dass die Parametervariierung gemäß einer Gleichverteilung zu erfolgen hat (DSM_Parameter_1, siehe Abschnitt 2.1.2).

Bei Parametern mit Listencharakter (z.B. Parameter_Liste_1) muss die Einheit am Ende der Liste angegeben werden, da diese vom Programm erwartet wird. Außerdem hilft die Angabe der Einheit „Uhr“, eine

Zeitangabe zu erkennen, welche eine spezielle Behandlung beim Einlesen benötigt, da EXCEL eine andere serielle Zeitnummer verwendet als MATLAB. Eine eventuelle Standardabweichung wird in der, den Werten nachfolgenden Zeile angegeben. Die Bezeichnung dieser Zeile kann willkürlich gewählt werden, im Beispiel ist diese „'-> Std.Dev. :“ um die Übersichtlichkeit zu erhöhen.

Alle in der Simulationssoftware verwendeten Parameter werden gemeinsam in der Funktion `get_default_values` definiert und können dort zentral eingesehen und bearbeitet werden. In den folgenden Abschnitten werden alle Parameter vorgestellt und genau auf ihre Eigenschaften eingegangen.

2.1.1. Simulationseinstellungen

Die Simulationseinstellungen legen die wichtigsten globalen Parameter für einen Simulationsdurchlauf fest. Sie können entweder direkt in einer Parameterdatei definiert und von dieser eingelesen (siehe Tabelle 2.1) oder über das GUI geändert werden (siehe Abschnitt 2.5). In Tabelle 2.2 sind die Einstellungen für einen Simulationsdurchlauf zusammengefasst.

Name	Beschreibung
Date_Start	Startzeitpunkt der Simulation. Als Datum in der Form dd.mm.yyyy HH:MM anzugeben
Date_End	Endzeitpunkt der Simulation. Als Datum in der Form dd.mm.yyyy HH:MM anzugeben
Sim_Resolution	Zeitliche Auflösung der Simulation bzw. Schrittweite der Simulationspunkte. Mögliche Auflösungen:
	sec 1 Sekunde
	min 1 Minute
	5mi 5 Minuten
	quh 15 Minuten
hou 1 Stunde	
Number_User	Anzahl der Personen, deren durchschnittlicher Verbrauch simuliert werden soll. Wird hier der Wert 1 angegeben, wird von jedem zu simulierenden Gerätetyp ein aktives Gerät erzeugt und simuliert (siehe Abschnitt 2.2.1, Attribut <code>Activity</code>)
Use_DSM	Wird hier der Wert 1 angegeben, werden alle Geräte mit ihrer DSM-Funktion simuliert, bei Wert 0 nur die Geräteinstanzen ohne DSM
Use_Same_DSM	Optional. Gibt an, ob bei jedem Simulationsdurchlauf versucht werden soll, die gleichen DSM-Device-Instanzen zu verwenden.
Device_Assembly	Gibt an, welche Gerätetypen für die Simulation herangezogen werden sollen. Wert 1 → Gerät verwenden, Wert 0 → Gerät nicht verwenden.

Tabelle 2.2: Zusammenfassung der Simulationseinstellungen

`Date_Start` und `Date_End` legen den Zeitraum fest, für den die Simulation durchgeführt werden soll, `Sim_Resolution` in welcher zeitlichen Auflösung dies zu erfolgen hat. Es kann aus fünf verschiedenen Auflösungen, welche in Tabelle 2.2 angeführt sind, ausgewählt werden.

`Number_User` legt die Anzahl an Personen fest, deren Geräte in das zu simulierende Kollektiv aufgenommen werden. Es können hier nur natürliche Zahlen angegeben werden. Wird hier der Wert 1 angegeben, wird nicht der durchschnittliche Gerätemix betrachtet, den eine Person hat, sondern es wird von jedem Gerät eine aktive Instanz erzeugt und simuliert. Das ist hilfreich bei der Betrachtung von Effekten, die ein einziges Gerät betreffen. Die Angabe von Personen hat gegenüber einer Angabe der Zahl von Geräten den Vorteil, dass nur einmal ein Wert für jede Gerätekonstellation angegeben werden muss. Alle weiteren Parameter und Funktionen des Simulationsprogramms sind dann dahingehend so ausgelegt, so viele Geräte zu erstellen, dass ein zu erzielender Ausstattungsgrad erreicht wird, der die reale Situation bzw. die Vorgaben widerspiegelt (siehe Kapitel 3). Über die Angabe der Geräteanzahl wäre dies nur viel aufwändiger zu realisieren. Auch können auf Basis der Zahl der Gerätebesitzer Vergleiche von unterschiedlichen Simulationen gemacht werden, indem man sich auf die Leistungsaufnahme pro Person bezieht.

Über `Device_Assembly` werden jene Gerätetypen angegeben, die für die Simulation in Betracht gezogen werden sollen. Im Prinzip handelt es sich dabei um eine Liste mit den verfügbaren und in der Parametertabelle definierten Gerätetypen, die mit der Angabe eines logischen Wertes ausgewählt werden. Auch können nicht benötigte Geräte aus dieser Liste einfach entfernt werden. Die Reihenfolge der Gerätetypen kann willkürlich gewählt werden.

Mit der Option `Use_DSM` wird festgelegt, ob die Geräte mit DSM simuliert werden. Durch Deaktivieren dieser Option durch Angabe des Wertes 0 wird nur die Leistungsaufnahme der Geräteinstanzen über die Simulationsdauer ermittelt, was einen deutlich geringeren Simulationsaufwand bedeutet. Dies kann bei der Parameterermittlung für die Gerätetypen hilfreich sein, da so unnötiger Rechenaufwand eingespart wird.

Die Option `Use_Same_DSM` kann optional angegeben werden. Sie wird an und für sich nur für die Verwendung bei Simulationsreihen (siehe Abschnitt 2.5.4) benötigt. Ist sie aktiv (Wert = 1) werden bei mehreren Simulationsdurchläufen, die aufeinander folgen, immer die gleichen Geräte mit den gleichen `DSM_Device`-Instanzen verwendet. So kann bei unterschiedlichen Simulationsauflösungen bzw. Simulationszeiträumen sichergestellt werden, dass es zu keiner Veränderung im verwendeten Gerätekollektiv kommt. Ist die Option deaktiviert, werden vor jedem Simulationsdurchlauf neue `DSM_Device`-Instanzen erzeugt (siehe Abschnitt 2.3).

2.1.2. Geräteeigenschaften

Um die Definition der Geräteeigenschaften so flexibel wie möglich zu gestalten, werden alle Parameter für einen Gerätetyp beim Einlesen der Parameterdatei zu einer Parameterliste zusammengefügt. Dies hat den Vorteil der Unabhängigkeit der Angaben vom Gerätetyp und von der Parameteranzahl, was bei einer starren Variablendefinition nicht möglich wäre. Erst bei der Instanzenerzeugung werden die Einträge in dieser Parameterliste individuell von jeder Geräteklasse überprüft und angewendet.

Die programminterne Parameterliste besteht immer aus Trippeln der Form

```
{Parameter_Name, Mean_Value, Standard_Deviation}
```

`Parameter_Name` gibt an, welcher Parameter übergeben wird. `Mean_Value` gibt den oder die Mittelwerte an, `Standard_Deviation` die Standardabweichung, welche normalerweise in Prozent des Mittelwerts angegeben wird. Ist `Mean_Value` eine oder mehrere Zeitangaben (Uhrzeit, Datum), wird die Standardabweichung als Zeitraum in Minuten angesehen. Die Angabe von `Parameter_Name` und `Mean_Value` ist zwingend erforderlich. Muss keine Standardabweichung angegeben werden, weil es sich z.B. beim angegebenen Parameter nur um eine Option mit boolescher Charakteristik handelt, wird hier automatisch der Wert 0 eingetragen.

In der Funktion `vary_parameter` werden normalerweise mit Hilfe der MATLAB-Funktion `normrnd` die übergebenen Parameter statistisch gemäß einer Normalverteilung gestreut. Dies erfolgt typischerweise bei der Erzeugung der jeweiligen Geräteinstanzen (siehe Abschnitt 2.2.1). Danach besitzt jede Instanz einer Geräteklasse einen individuellen Satz von Parameterwerten. Mit diesen Parameterwerten wird dann in der jeweiligen Klasse die Funktionalität des Gerätes bestimmt, wodurch es zu scheinbar unabhängigen, individuellen Verhaltensweisen kommt. Von einer Normalverteilung kann hier bei den meisten Parametern aufgrund der großen Anzahl von weitgehend unabhängigen Zufallsvariablen ausgegangen werden [13: S.4]. Die Normalverteilung bei der Parametervariation ist sehr gut im Diagramm 2.13 auf Seite 26 zu erkennen.

An dieser Stelle sei auf eine Besonderheit bei der Parametervariierung hingewiesen. Bei einer hohen Streuung und passendem Mittelwert kann es vorkommen, dass negative Parameterwerte ermittelt werden. Diese machen aber oft keinen Sinn, wie z.B. ein negativer Wert des Parameters `Time_Delay_Restore_Op`, welcher eine Zeitspanne definiert (siehe Abschnitt 2.3.1.2.2). Ein negativer Wert würde einer negativen Zeitspanne entsprechen und hier bedeuten, dass manche DSM-Einheiten bereits vor einer zukünftigen Erholung der Netzfrequenz von dieser wissen, was aber kausal nicht zu erklären wäre. Im Beispiel von Diagramm 2.1 ist dies anhand der punktierten Linie dargestellt: Um 8:30 Uhr sind bereits einige Geräte wieder aktiv, obwohl alle DSM-Einheiten in diesem Beispiel erst ab einer Netzfrequenz von 49 Hz die Leistungsreduktion beenden sollten.

In Diagramm 2.1 sind verschiedene Möglichkeiten dargestellt, wie negative Parameterwerte behandelt werden. Am einfachsten ist das Nullsetzen der negativen Parameter (blaue Linie). Dies führt aber zu einem vermehrten Auftreten des Wertes „0“ und im Beispiel dazu, dass viele Geräte gleichzeitig ihre Leistungsreduzierung beenden, was in diesem Fall nicht gewünscht ist. Bei vielen anderen Parametern ist diese Vereinfachung jedoch ausreichend.

Weiters kann nach der Streuung der Betrag des Parameters herangezogen werden (grüne Kurve).

Verwendet wird jedoch bei kritischen Parametern das Verfahren, dass im Fall eines negativen Wertes der Parameter solange gestreut wird, bis ein positiver Wert ermittelt wird (rote Kurve). Dies führt dazu, dass die Glockenkurve der Normalverteilung „verbogen“ wird, so dass sie kontinuierlich zum Wert Null hin ausklingt. Es liegt aber in diesem Fall keine Normalverteilung der Parameterwerte mehr vor.

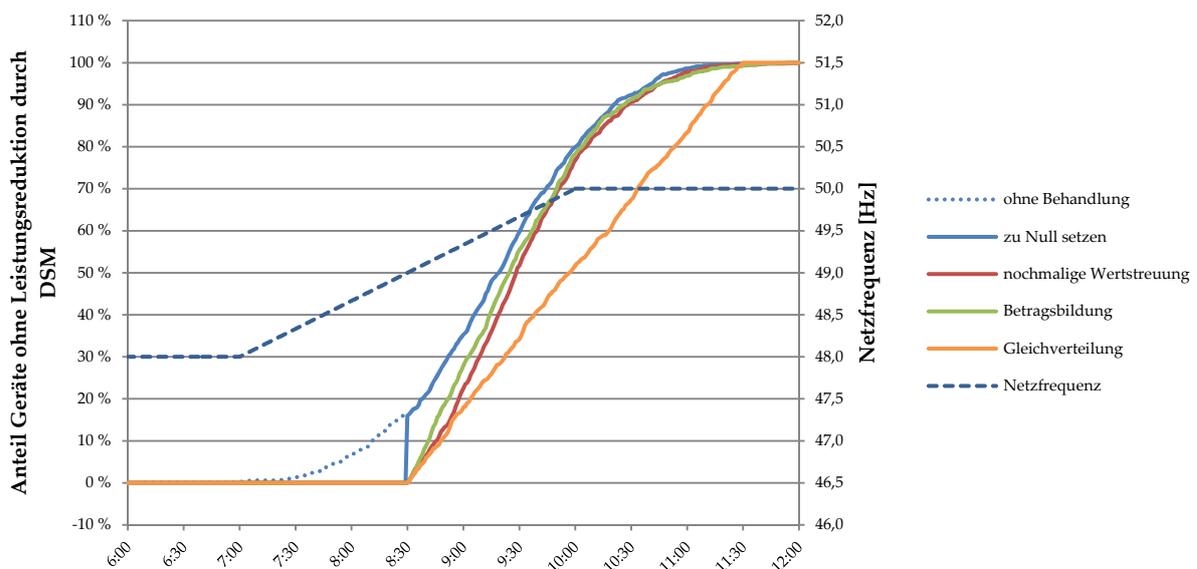


Diagramm 2.1: Auswirkung bei der Behandlung negativer Parameterwerte am Beispiel des Parameters
`Time_Delay_Restore_Op = 100 min, Std.Dev. = 50%`

Es sollte daher bei der Wahl der Standardabweichungen darauf geachtet werden, dass diese nicht allzu groß angenommen werden. Bei einer Standardabweichung, die kleiner als 33% vom Mittelwert ist, sind 99,7% der Parameterwerte positiv, weshalb sich ev. Verzerrungen der Normalverteilung nicht allzu stark auswirken. Höhere Werte für die Standardabweichung sollten nur in Ausnahmefällen verwendet werden.

Bei manchen Parametern ist eine Normalverteilung nicht zielführend, weshalb auch die Parametervariierung gemäß einer Gleichverteilung möglich ist. Dazu muss für den Wert von `Standard_Deviation` der String `'Uniform_Distr'` angegeben werden. Dann werden alle Parameterwerte zwischen den Werten 0 und `Mean_Value` gleichverteilt, zu sehen in Diagramm 2.1 (gelbe Kurve).

Die einzelnen Parameter, die für die unterschiedlichen Gerätetypen angegeben werden können, werden im Abschnitt 2.2 genauer erläutert.

2.1.3. DSM-Funktion

Die Parameterangabe für die Eigenschaften der DSM-Funktionen erfolgt sehr ähnlich wie bei den Geräteparametern. Es wird genauso aus den in der Parameterdatei angegebenen Daten eine Parameterliste erzeugt, die dann bei der Instanzenerzeugung abgearbeitet wird.

Einzigster Unterschied liegt in der Möglichkeit der Angabe unterschiedlicher Moden der DSM-Funktionen sowie deren Verteilung über die einzelnen `DSM_Device`-Instanzen (siehe Beispiel in Tabelle 2.1, z.B. `„DSM_Input_Mode“`). Die Angabe der Namen und deren Verteilung hat so wie in Tabelle 2.1 zu erfolgen, jedoch dienen die zusätzlichen „Pfeile“ nur der besseren Übersicht. Die Namen der Moden müssen mit den im Programm definierten Bezeichnungen übereinstimmen, da sonst keine Zuordnung erfolgen kann. Für jeden der angegebenen Moden müssen alle zugehörigen Parameter definiert werden, da erst bei der Instanzenerzeugung festgelegt wird, welcher Modus bei der Simulation in der jeweiligen Instanz verwendet werden wird.

Eine genauere Erläuterung der einzelnen Parameter sowie deren Eigenschaften und Auswirkungen erfolgt im Abschnitt 2.3.

2.2. Simulation der Verbraucher

Um die Auswirkungen von DSM auf Verbraucher und deren Leistungsaufnahme zu untersuchen, wurde versucht, Verbraucher in ihrem Verhalten nachzubilden, damit deren Reaktion auf verschiedene Situationen untersucht werden kann.

Zu diesem Zweck wurden mehrere Verbrauchergruppen definiert, die die verschiedenen Charakteristika der Mitglieder der jeweiligen Gruppen repräsentieren. Für das Implementieren der Verbrauchergruppen

wurde der naheliegende objektorientierte Ansatz gewählt. Jedes Gerät ist eine Instanz seiner zugehörigen Geräteklasse. Die Definition der einzelnen Geräteklassen leitet sich aus den Eigenschaften und dem Verhalten der jeweiligen Gerätegruppen ab. Im Folgenden werden die wichtigsten Eigenschaften der implementierten Geräteklassen genauer erläutert.

2.2.1. Device

Alle anderen Geräteklassen sind Subklassen der Geräteklasse `Device`, welche alle Eigenschaften zusammenfasst, die Elektrogeräte gemeinsam haben. In Abbildung 2.1 ist grob die Struktur der Geräteklassen skizziert.

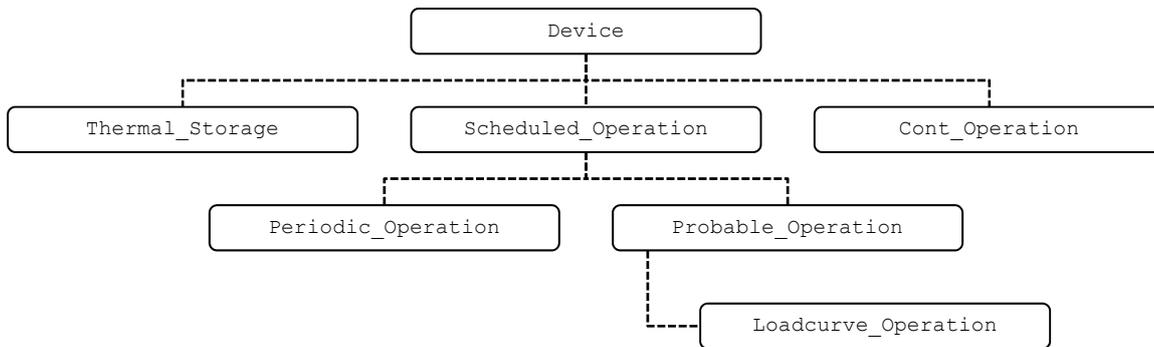


Abbildung 2.1: grobe Übersicht über die Struktur der implementierten Geräteklassen.

Die wichtigste Eigenschaft der Geräteklasse `Device` liegt darin, dass diese die Parameter für ein Gerät anhand der übergebenen Argumenten-Liste gemäß einer Normalverteilung streut, damit jede Instanz eines Gerätes individuelle Eigenschaften aufweist (siehe Abschnitt 2.1.2).

	Name	Einheit	Beschreibung
Parameter	Power_Nominal	W	Anschlussleistung des Geräts
	Start_Probability	%	Wahrscheinlichkeit, dass Gerät aktiv wird. Kann eine, zu einer Startzeitliste gehörende Liste sein (definiert dann für jeden Startzeitpunkt die Wahrscheinlichkeit, ob Gerät aktiv wird), oder auch ein Wert, der die Wahrscheinlichkeit für die generelle Aktivität angibt (für die gesamte Simulationsdauer).
Attribute	Activity	-	Ist das Gerät irgendwann im Einsatz? (Nach Erzeugen der Geräteinstanzen können so alle nichtaktiven Geräte aussortiert werden. Daher sollte bei den im Simulationsprogramm verwendeten Geräteinstanzen immer <code>Activity=1</code> sein!)
	Operating	-	Gibt an, ob das Gerät gerade aktiv ist (d.h. eingeschaltet)
	DSM	-	Instanz der Klasse <code>DSM Device</code> , welche das DSM-Verhalten des Verbrauchers beinhaltet und steuert (siehe Abschnitt 2.3)
Ausgabe	Power_Input	W	Leistungsaufnahme des Geräts zum aktuellen Zeitpunkt.

Tabelle 2.3: Übersicht der Properties der Klasse `Device`

Methode	Beschreibung
<code>check_activity</code>	<code>OBJ = CHECK_ACTIVITY(OBJ)</code> überprüft, ob Gerät überhaupt für Simulation als aktiv gilt und setzt dementsprechend <code>OBJ.ACTIVITY</code> . Dieser Wert hilft der das Gerät verwendenden Funktion zu entscheiden, ob dieses Gerät überhaupt für den Simulationsdurchlauf gespeichert werden soll oder einfach ignoriert wird (da es keinen Beitrag zum Gesamtergebnis liefert). Diese Funktion wird in den jeweiligen Subklassen genauer definiert!
<code>adapt_for_simulation</code>	<code>OBJ = ADAPT_FOR_SIMULATION(OBJ, VARARGIN)</code> sorgt dafür, dass die aktuelle Geräteklasse für einen Simulationsdurchlauf vorbereitet wird. Diese Funktion muss einmal zu Beginn der Simulation für jede Geräteinstanz aufgerufen werden. Diese Funktion wird in den jeweiligen Subklassen genauer definiert!
<code>next_step</code>	<code>OBJ = NEXT_STEP(OBJ, TIME)</code> ermittelt die Reaktion der Geräteinstanz zum Zeitpunkt <code>TIME</code> . Die Reaktion besteht vordergründig in der aufgenommen Leistung zu diesem Zeitpunkt. Diese Funktion wird in den jeweiligen Subklassen genauer definiert!

Tabelle 2.4: Übersicht über die Methodenrumpfe, die in der Klasse `Device` definiert werden, und in allen Geräteklassen implementiert werden müssen.

Eine Übersicht über die Eigenschaften der Geräteklasse `Device` gibt Tabelle 2.3. Die darin mit „Parameter“ bezeichneten Zeilen werden der Klasse von „außen“ (durch die Parameterdefinition) übergeben, die mit „Attribute“ bezeichneten Zeilen stellen klasseninterne Werte dar, die während der Erzeugung der Instanz bzw. während der Simulation des Gerätes verwendet werden. Weiters werden in der Geräteklasse `Device`

jene Methoden als Funktionsrümpfe definiert, welche alle Geräteklassen besitzen müssen, damit sie in der Simulation eingesetzt werden können. In Tabelle 2.4 sind diese Methoden angeführt.

Die wichtigste Funktion für das Simulationsprogramm ist die Methode `next_step`. In dieser Funktion wird vordergründig ermittelt, welche Leistung das Gerät zu einem bestimmten Zeitpunkt hat. Je nach Gerätekategorie und Gerätefunktion wird dieser Wert unterschiedlich ermittelt.

Ansonsten ist in dieser Gerätekategorie keine weitere wichtige Funktionalität zu finden, da diese Klasse nur den Rahmen für die nachfolgenden Subklassen liefert.

2.2.2. Cont_Operation

Das einfachste Verhalten weist die Gerätekategorie `Cont_Operation` auf. Diese Gerätekategorie repräsentiert alle Geräte, die eine zeitunabhängige, konstante Leistungsaufnahme aufweisen.

Diese Gerätekategorie hat dieselben Parameter und Attribute wie die Superklasse `Device`. Die Funktion `next_step` gibt nur den Wert der Anschlussleistung `Power_Nominal` zurück.

Diese Gerätekategorie wurde ursprünglich dazu verwendet, um sämtliche Stand-by-Verbraucher zusammenzufassen. Da aber Geräte, die im Betrieb sind, keinen Stand-by-Verbrauch aufweisen, wurde dazu übergegangen, den Stand-by-Verbrauch den jeweiligen Gerätekategorien direkt zuzuordnen, da nur so dieses Verhalten nachgebildet werden kann. Damit erübrigte sich die Notwendigkeit der Verwendung dieser Klasse in der Simulation, da sonst keine anderen Verbraucher mit diesem Verhalten simuliert werden.

Beim Testen der implementierten DSM-Input-Funktionen (siehe Abschnitt 2.3.1) wurde diese Klasse dennoch häufig verwendet, weil mit ihr eine einfache Visualisierung der Reaktion der `DSM_Device`-Instanzen möglich ist.

2.2.3. Thermal_Storage

Die Klasse `Thermal_Storage` soll alle thermischen Speicher charakterisieren, deren thermischer Energieinhalt mit Hilfe von elektrischer Energie manipuliert wird. Beispiele dafür sind Kühlschränke, Warmwasserboiler, Raumheizungen, Klimaanlage, Wärmepumpen usw.

2.2.3.1. Model eines thermischen Speichers

Um eine große Klasse an möglichen Verbrauchern abzudecken, werden die Eigenschaften und Methoden dieser Klasse sehr allgemein formuliert. Ausgegangen wird von einem sehr einfachen Modell, worin der Speicher eine thermische Kapazität besitzt, die mit Hilfe einer Isolierung, welche einen thermischen Widerstand darstellt, von der „Umgebung“ abgeschirmt wird, dargestellt in Abbildung 2.2.

Abbildung 2.3 zeigt dieses Modell, in dem die thermischen Größen durch ihre elektrischen Äquivalente ersetzt wurden.

Die dem System zugeführte thermische Leistung P_Z sowie die thermischen Verluste P_V sind zum elektrischen Strom äquivalent. Die Umgebungstemperatur ϑ_A entspricht einer Spannung, der thermische Widerstand und die thermische Kapazität ihren äquivalenten elektrischen Größen. Die innere Temperatur des thermischen Speichers ϑ entspricht der Spannung an der Kapazität.

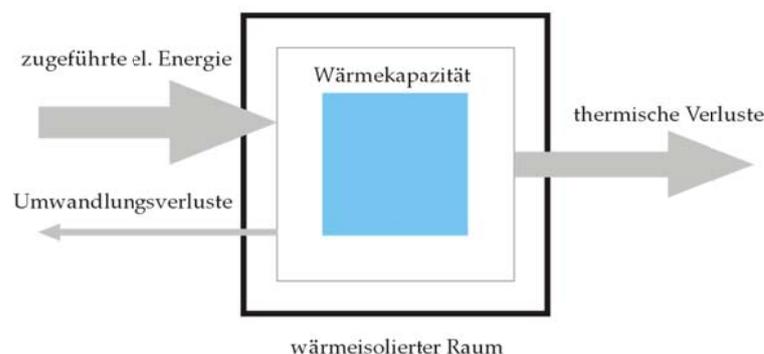


Abbildung 2.2: Modell für die Klasse `Thermal_Storage` (nach [9: S.47])

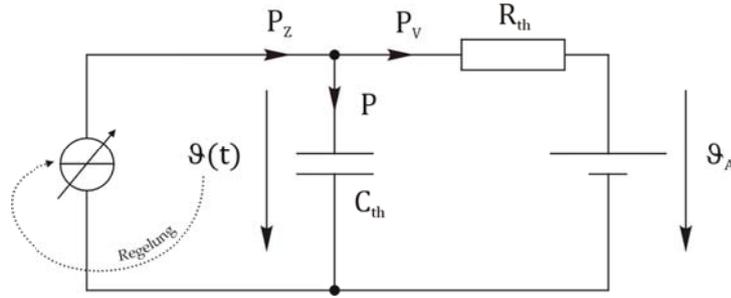


Abbildung 2.3: Modell eines thermischen Speichers, dargestellt als elektrische Schaltung (nach [9: S.46])

Die dem thermischen Speicher in Summe zugeführte thermische Leistung P setzt sich aus der zugeführten Leistung sowie den Verlusten bzw. aus der Änderung der Temperatur im Speicher sowie dessen thermischer Kapazität folgendermaßen zusammen:

$$P = P_Z - P_V = C_{th} \frac{d\vartheta(t)}{dt} \quad (\text{vgl. } i_C = C \frac{du_C}{dt}) \quad (2.1)$$

Die Verluste sind abhängig von der Temperaturdifferenz zwischen Speicher und Umgebung sowie der Qualität der thermischen Isolierung des Speichers, repräsentiert durch den thermischen Widerstand R_{th} :

$$P_V = \frac{1}{R_{th}} (\vartheta(t) - \vartheta_A) \quad (\text{vgl. } i_R = \frac{u_R}{R}) \quad (2.2)$$

P_Z ist vom Zustand und der Betriebsweise des Speichers abhängig. Je nach der Temperatur im Speicher wird, um die Verluste auszugleichen, thermische Energie zu- oder abgeführt, eine dementsprechende Regelung bzw. Steuerung vorausgesetzt. Erfolgt dies über eine elektrische Umwandlung (z.B. Wärmepumpe), ist diese eventuell mit einem Wirkungsgrad η behaftet. Weiters werden alle thermischen Größen auf die elektrische Seite des thermischen Speichers bezogen, um so den Einfluss der Leistungszahl einer eventuell vorhandenen Wärmepumpe nicht berücksichtigen zu müssen.

Bei Heizgeräten wird thermische Energie dem Speicher hinzugefügt, was durch ein positives Vorzeichen der elektrischen Leistung berücksichtigt wird:

$$P_{On, \text{heizen}} = +\eta \cdot P_{el} \quad (2.3)$$

Bei Kühlgeräten wird thermische Energie dem Speicher entzogen, was dementsprechend einem negativen Vorzeichen entspricht:

$$P_{On, \text{kühlen}} = -\eta \cdot P_{el} \quad (2.4)$$

Allgemein ausgedrückt mit Einführung der Richtung des Flusses der thermischen Nutz-Energie d_{therm} :

$$d_{therm} := \begin{cases} -1 & \text{für kühlen} \\ +1 & \text{für heizen} \end{cases} \quad (2.5)$$

$$P_{On} = d_{therm} \cdot \eta \cdot P_{el} \quad (2.6)$$

Im Folgenden wird eine simple Thermostat-Regelung des thermischen Speichers angenommen. Diese Art der Temperaturregelung ist bei den hier nachzubildenden Geräten sehr weit verbreitet, weshalb diese Wahl keine gravierende Einschränkung der Verwendbarkeit dieser Geräteklasse darstellt. Bei dieser Regelung wird eine Solltemperatur ϑ_{Set} und eine Schaltschwelle ϑ_{SP} definiert, innerhalb der sich die Temperatur des Speichers bewegen kann. Für Kühlgeräte ergibt sich dann beispielsweise [9]:

$$P_Z(\vartheta) = \begin{cases} -\eta \cdot P_{el} & \text{wenn } \vartheta > \vartheta_{Set} + \frac{\vartheta_{SP}}{2} \\ 0 & \text{wenn } \vartheta < \vartheta_{Set} - \frac{\vartheta_{SP}}{2} \\ \text{vorhergehender Zustand} & \text{sonst} \end{cases} \quad (2.7)$$

Damit folgt für die Änderung der Temperatur des thermischen Speichers aus (2.1) und (2.2):

$$\frac{d\vartheta(t)}{dt} = \frac{1}{C_{th}} \left(P_Z(\vartheta) - \frac{(\vartheta(t) - \vartheta_A)}{R_{th}} \right) = \frac{1}{R_{th} C_{th}} (P_Z(\vartheta) \cdot R_{th} + \vartheta_A - \vartheta(t)) \quad (2.8)$$

Durch Lösen dieser Differentialgleichung kann die die aktuelle Temperatur im Speicher ermittelt werden.

2.2.3.2. Simulation des thermischen Speichers

Für die Simulation wird die Differentialgleichung (2.8) durch das Einführen diskreter Zeitschritte vereinfacht.

$$\frac{\Delta\vartheta}{\Delta t} = \frac{1}{R_{th}C_{th}} (P_Z(t - \Delta t) \cdot R_{th} + \vartheta_A - \vartheta(t - \Delta t)) \quad (2.9)$$

$P_Z(t - \Delta t)$ ist die thermische Leistung, die dem Speicher seit dem letzten Simulationsschritt zugeführt wird. $\vartheta(t - \Delta t)$ ist die Temperatur des Speichers beim letzten Simulationsschritt. Durch die Vereinfachung wird erreicht, dass nun pro Simulationsschritt nur mehr die Änderung der Temperatur berechnet und diese Änderung zu der Temperatur des vorhergehenden Schrittes addiert werden muss, um die aktuelle Temperatur im Speicher zu erhalten:

$$\Delta\vartheta = \frac{1}{R_{th}C_{th}} (P_Z(t - \Delta t) \cdot R_{th} + \vartheta_A - \vartheta(t - \Delta t)) \cdot \Delta t \quad (2.10)$$

$$\vartheta(t) = \vartheta(t - \Delta t) + \Delta\vartheta \quad (2.11)$$

Anhand der aktuellen Temperatur wird der aktuelle Betriebszustand und damit $P_Z(t)$ ermittelt, woraus sich die aufgenommene elektrische Leistung zum Zeitpunkt t ergibt.

Wenn die Zeitschritte Δt klein genug gegenüber der thermischen Zeitkonstante des thermischen Speichers gewählt werden, ist der Fehler, der durch die Vereinfachung entsteht, gering. Aus diesem Grund wird eine maximale Dauer eines Simulationsschrittes in der Klasse `Thermal_Storage` von $\Delta t \leq 60$ Sekunden zugelassen. Dieser Wert wurde gewählt, weil damit Kühlgeräte wie Kühlschränke noch gut nachgebildet werden können, welche die Geräte mit der kleinsten Zeitkonstante darstellen (im Vergleich zu Boilern oder Raumheizungen) und sich mit diesem Wert der Aufwand der Berechnung in Grenzen hält. Denn je kleiner Δt gewählt wird, desto mehr Rechenschritte werden benötigt.

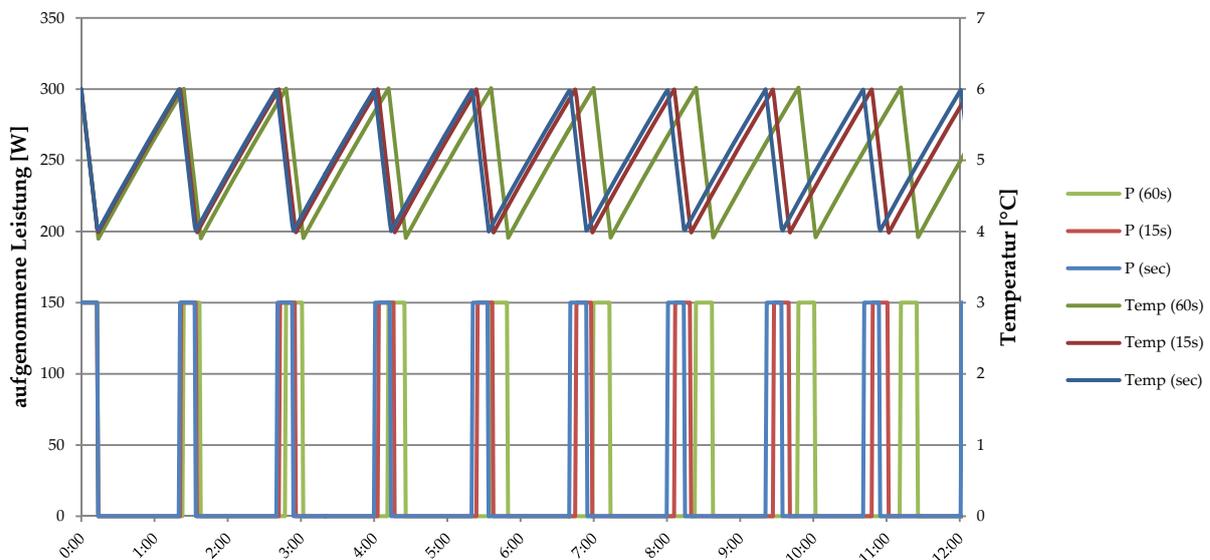


Diagramm 2.2: Vergleich des gleichen thermischen Speichers (Kühlgerät) mit unterschiedlichen Werten für Δt

Diagramm 2.2 zeigt den Unterschied beim Simulationsergebnis mit verschiedenen Werten von Δt . Man sieht, dass der qualitative Verlauf ein sehr ähnlicher ist, aufgrund der größeren Auflösung der Schaltzeitpunkte wird aber der Betriebszustandswechsel immer etwas zu spät durchgeführt, wodurch sich die etwas zu langen Zeiträume zwischen den Aktivitätszeiten erklären lassen. Die Dauer, in der dem Speicher thermische Energie zugeführt (hier im Beispiel eigentlich entzogen) wird, ist aber, abgesehen von einer gleichbleibenden Abweichung im Minutenbereich, konstant.

Diagramm 2.3 verdeutlicht diesen Effekt. Wird der thermische Speicher über längere Zeit keinem Betriebszustandswechsel unterworfen, stimmen die Simulationsergebnisse weitgehend miteinander überein. Erst wenn wieder Zustandswechsel erfolgen, kommt es zu immer größer werdenden Abweichungen der einzelnen Simulationsergebnisse voneinander (in Diagramm 2.3 ab 12:00 Uhr). Diese Abweichungen können aber korrigiert werden, was im Abschnitt 2.2.3.3 genauer erläutert wird.

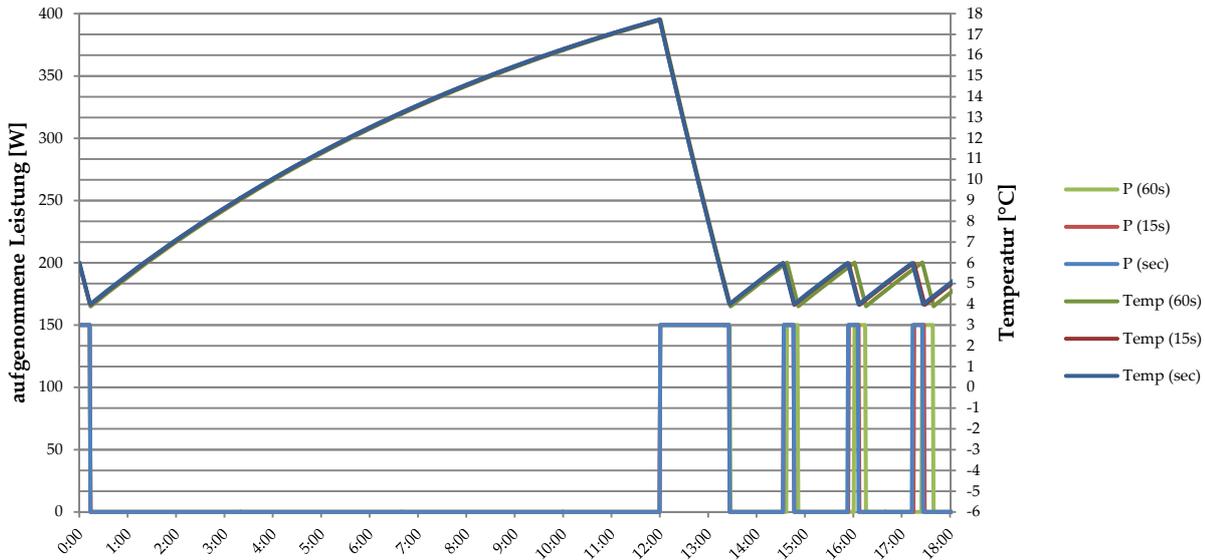


Diagramm 2.3: Vergleich des gleichen thermischen Speichers (Kühlgerät) mit unterschiedlichen Werten für Δt und Inaktivität des Geräts von 0:30 bis 12:00 Uhr

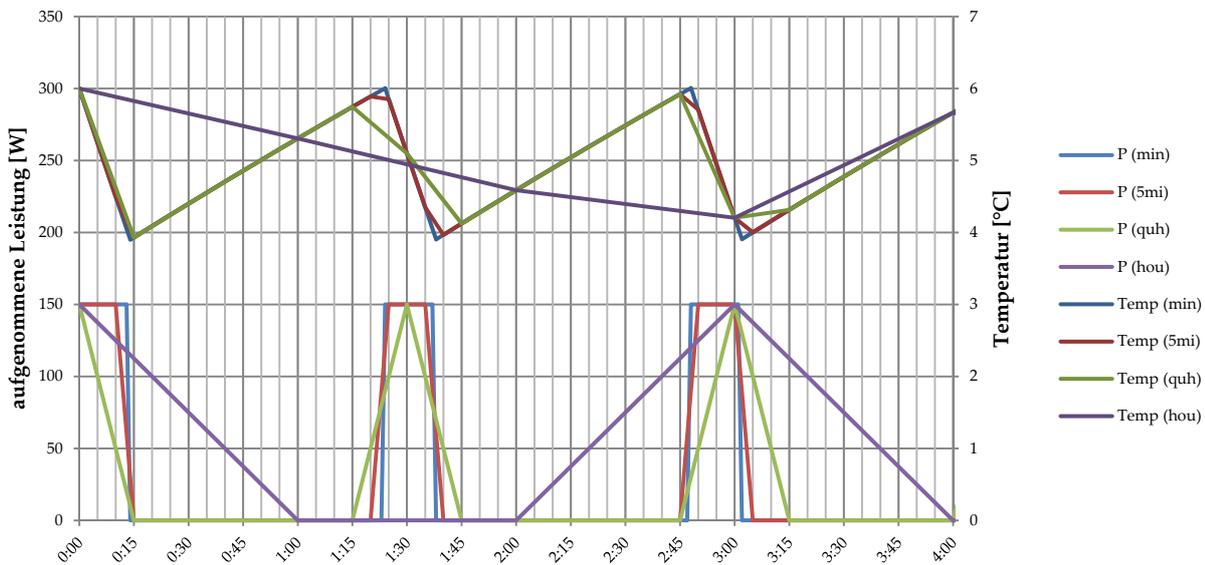


Diagramm 2.4: Vergleich der aufgenommenen Leistung und der Temperatur eines thermischen Speichers (Kühlgerät) für verschiedene globale Simulationsauflösungen.

Ist die Dauer der globalen Simulationsschritte (*Sim_Resolution*, siehe 2.1.1) größer als 60 Sekunden, werden für die Ermittlung der Temperatur und des Betriebszustands so viele Zwischenschritte eingefügt, dass sich für $\Delta t = 60$ Sekunden ergeben. Dadurch wird gewährleistet, dass unabhängig von der Simulationsauflösung die gleichen Werte zu den gleichen Zeitpunkten ermittelt werden. Diagramm 2.4 zeigt diese Übereinstimmung. Durch die Zwischenschritte schneiden sich alle Simulationsergebnisse zur vollen Stunde, was der größten Auflösung entspricht. Gut ist auch beim Verlauf der 5mi-Auflösung die Übereinstimmung alle 5 Minuten mit den Werten des Simulationsdurchlaufes in min-Auflösung zu erkennen.

2.2.3.3. Parameter und Auswirkung auf das Simulationsergebnis

In Tabelle 2.5 sind die wichtigsten Parameter und Attribute der Klasse *Thermal_Storage* aufgelistet. Die Parameter entsprechen alle, bis auf *Operat_Sim_Start* und *Start_Probability*, den Parametern des thermischen Modells.

Start_Probability ermöglicht den Ausstattungsgrad mit Geräten dieser Klasse zu steuern, indem der Prozentsatz für einen Einsatz angegeben wird. Da für jede Person eine Geräteinstanz erzeugt wird, kann dieser Wert gleich mit der Anzahl von Geräten pro Kopf gesetzt werden, wodurch die richtige Anzahl an Geräten in den Gerätesatz für die Simulation aufgenommen wird (siehe Abschnitt 3.5).

Um zu Beginn eines Simulationsdurchlaufes keine starken Schwankungen bei der Leistungsaufnahme der thermischen Speicher zu haben, weil sich z.B. alle Instanzen der Geräte gleichzeitig einschalten, werden während der Instanzenzeugung die Werte der Temperatur in den Speichern (Temp) zufällig gemäß einer Gleichverteilung in den Bereich zwischen den Schaltschwellen (Switch_Point) gesetzt. Dann wird mit Hilfe des Parameters Operat_Sim_Start ermittelt, ob die Zufuhr an thermischer Energie gerade aktiv oder inaktiv ist (aktueller Betriebszustand Operating). Operat_Sim_Start gibt an, wie viele der Speicher zum Startzeitpunkt aktiv sein müssen. Die Höhe dieses Wertes richtet sich nach der durchschnittlichen Periodendauer und Laufzeit der Gerätegruppe. Wenn z.B. diese im Schnitt alle 100 min für 10 min aktiv ist, muss der Wert von Operat_Sim_Start auf 10% gesetzt werden, da auch zu allen weiteren Zeitpunkten davon ausgegangen werden kann, dass im Schnitt 10% der Geräte aktiv sind. Mit diesen beiden Maßnahmen wird erreicht, dass zum Simulationsstart jeder Speicher in einem gültigen Betriebszustand ist und es zu keinen gravierenden Abweichungen der Leistungsaufnahme zum Startzeitpunkt kommt.

	Name	Einheit	Beschreibung
Parameter	Power_Nominal	W	Anschlussleistung des Geräts
	Start_Probability	%	Wahrscheinlichkeit, dass Gerät aktiv wird.
	Dir_therm_Flow	-	Richtung des Flusses der thermischen Nutz-Energie in den Speicher bei normalere "Auf-ladefunktion": +1 = "Aufheizen" (z.B. Boiler, Raumheizung, ...) -1 = "Abkühlen" (z.B. Kühlschrank, Klimaanlage, ...)
	Efficiency	%	Wirkungsgrad der Umwandlung von elektrischer in thermische Energie
	Switch_Point	°C	Schaltpunkt um Solltemperatur in °C (Thermostatregelung)
	Heat_Capacity	J/K	Wärmekapazität, bezogen auf die elektrische Einspeisung
	Thermal_Res	K/W	thermischer Widerstand der Isolierung, bezogen auf die elektrische Einspeisung
	Temp_Set	°C	Solltemperatur
	Temp_Ambiance	°C	Umgebungstemperatur
	Operat_Sim_Start	%	Wahrscheinlichkeit, dass Gerät zu Beginn der Simulation aktiv ist
Attribute	Activity	-	Ist das Gerät irgendwann im Einsatz? (Nach Erzeugen der Geräteinstanzen können so alle nichtaktiven Geräte aussortiert werden. Daher sollte bei den im Simulationsprogramm verwendeten Geräteinstanzen immer Activity = 1 sein!)
	Operating	-	Gibt an, ob das Gerät gerade aktiv ist (d.h. eingeschaltet)
	DSM	-	Instanz der Klasse DSM Device, welche das DSM-Verhalten des Verbrauchers beinhaltet und steuert (siehe Abschnitt 2.3)
	Temp	°C	aktuelle Temperatur
Ausgabe	Power_Input	W	Leistungsaufnahme des Geräts zum aktuellen Zeitpunkt.

Tabelle 2.5: Übersicht der Properties der Klasse Thermal_Storage

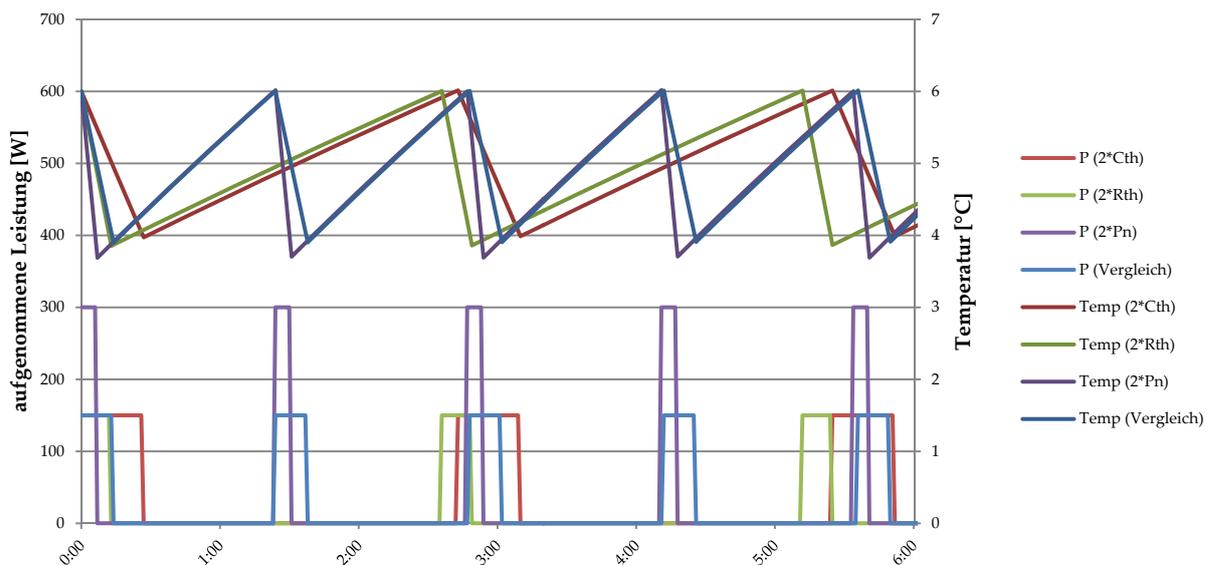


Diagramm 2.5: Auswirkung der Änderung verschiedener Parameter (Verdopplung) der Klasse Thermal_Storage am Beispiel eines Kühlgerätes

Diagramm 2.5 und Diagramm 2.6 zeigen, wie sich die Änderung verschiedener Parameter auf das Simulationsergebnis eines Kühlgerätes auswirken.

Wird C_{th} (Heat_Capacity, rote Kurven) verdoppelt, verdoppelt sich auch die Periodendauer der Leistungszufuhr, bei einer Halbierung von C_{th} halbiert sich diese. Das ist naheliegend, da sich die thermische Zeitkonstante aus dem Produkt von C_{th} und R_{th} zusammensetzt, weshalb sich eine Variierung von R_{th} (Thermal_Res, grüne Kurven) ähnlich auf die Periodendauer auswirkt.

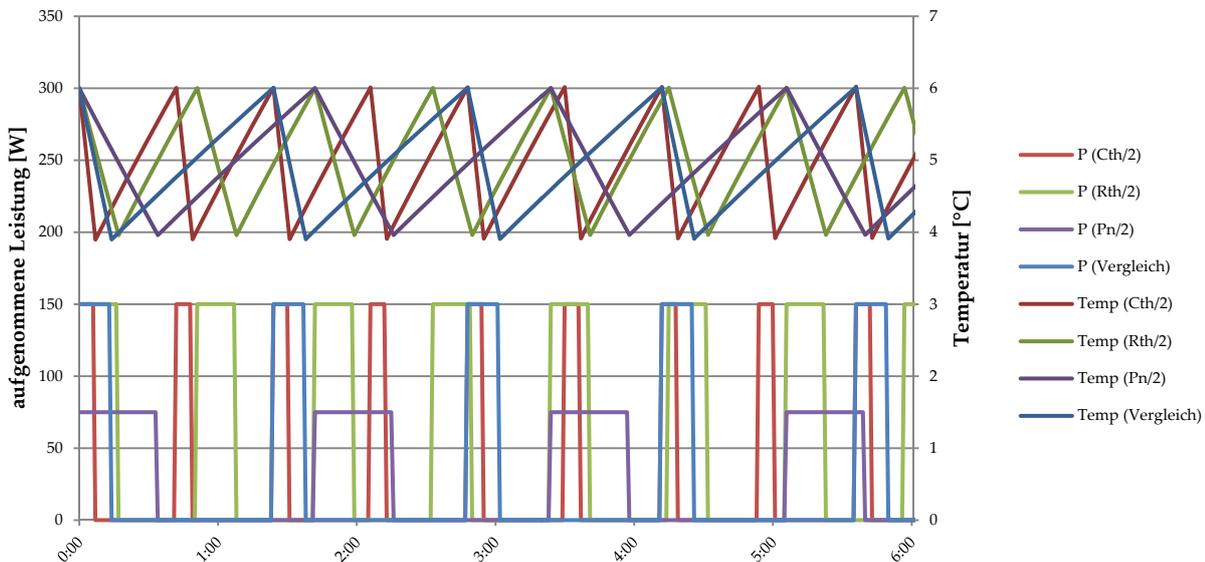


Diagramm 2.6: Auswirkung der Änderung verschiedener Parameter (Halbierung) der Klasse `Thermal_Storage` am Beispiel eines Kühlgerätes

Bemerkenswert ist, dass sich eine Variierung von R_{th} scheinbar kaum auf die Zeit auswirkt, in der von der oberen auf die untere Temperaturschranke gekühlt werden muss (gut zu erkennen zu Beginn der Diagramme), wohingegen die Variierung von C_{th} dort deutlich zu tragen kommt. Das lässt sich anhand von Gleichung (2.8) erklären: Die Wirkung der zugeführten Leistung P_Z ist unabhängig von R_{th} aber abhängig von C_{th} . Eine Verdopplung von C_{th} bedeutet, dass für das Abkühlen bei gleicher abgeführter thermischer Leistung doppelt so viel Zeit benötigt wird, was auch dem thermischen Modell entspricht: Eine doppelte Wärmekapazität heißt, dass bei gleicher Temperaturdifferenz doppelt so viel Wärmeenergie abgegeben werden muss, wodurch sich bei gleichbleibender Kühlleistungszufuhr die Dauer der Zufuhr ebenso verdoppeln muss. Ist das Gerät aber doppelt so gut isoliert, ändert das nichts an der nötigen Wärmeenergie, sofern die Verluste, die von R_{th} abhängen, klein gegenüber der zugeführten Kühlleistung sind. Die thermischen Verluste sind auch der Grund, weshalb sich die Variierung von R_{th} doch minimal auf die Dauer der Energiezufuhr auswirken.

Eine Variierung der elektrischen Anschlussleistung P_{el} (`Power_Nominal`, violette Kurven) wirkt sich hingegen wirklich nur auf die Dauer der Energiezufuhr aus, in der Erwärmungsphase gleichen sich die Verläufe (Parallelität der blauen und violetten Temperaturkurven), da die Dauer dieser Phase nur durch die thermische Zeitkonstante bestimmt wird.

In Diagramm 2.5 zeigen sich auch deutlich die Grenzen der gewählten Vereinfachung für die Simulation: Ist die Temperaturänderung zu stark, wird aufgrund der großen Zeitabstände ein deutlich größerer Fehler gemacht, was sich hier in einem deutlichen Unterschreiten der Schaltschwelle bei 4°C äußert.

In Diagramm 2.2 wurde das aktuelle Simulationsmodell mit einem genaueren Modell verglichen. Das größere Simulationsmodell scheint einen höheren thermischen Widerstand aufzuweisen. Diese virtuelle Erhöhung begründet sich aber nur in der größeren zeitlichen Auflösung.

Mit Hilfe der Erkenntnis, dass sich eine Variierung von R_{th} , wenn diese nicht allzu hoch ausfällt, kaum auf die Dauer der Abfuhr thermischer Energie auswirkt, kann man versuchen, die scheinbar höhere thermische Zeitkonstante des größeren Simulationsmodells so anzupassen, dass die Ergebnisse besser übereinstimmen.

Im Beispiel von Diagramm 2.2 konnte durch die Reduzierung des thermischen Widerstands um 4,5% eine deutliche Verbesserung erreicht werden (siehe Diagramm 2.7).

Diese Erkenntnis wird in Abschnitt 3.5 verwendet, um die Abweichungen des Simulationsmodells von errechneten Werten zu korrigieren.

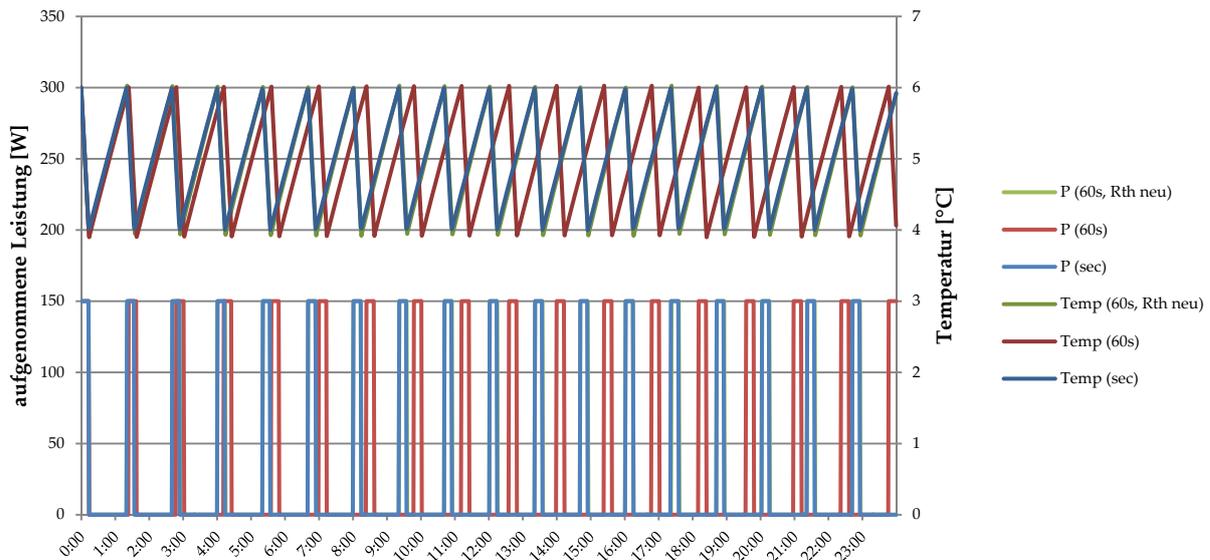


Diagramm 2.7: Erreichen der Übereinstimmung der Ergebnisse des fein und grob zeitlich aufgelösten thermischen Modells mit Hilfe von Variierung von R_{th}

2.2.3.4. Abweichungen Modell – Realität

In der vorliegenden Ausführung der Geräteklasse `Thermal_Storage` wurde für die gesamte Zeit der Aufnahme elektrischer Energie ein konstanter Wert der Leistung angenommen. Wie Messungen an einem Kühlschrank aber zeigen, weisen gerade diese Geräte einen deutlichen Einschaltstromstoß und damit erhöhte Leistungsaufnahme auf (siehe Diagramm 3.5 auf Seite 69 und [14: S.51]).

Die Ursache darin liegt im Anlaufstrom des Motors für den Kompressor. Um dies zu berücksichtigen könnte z.B. ein zusätzlicher, zeitabhängiger, nach einer exponentiellen Kurve abfallender Leistungsterm definiert werden, der durch seine Starthöhe und Zeitkonstante definiert ist. Dieser Leistungsterm könnte dann zu der normalen Leistung dieser und eventuell anderer Geräteklassen, die ein ähnliches Verhalten aufweisen, hinzugefügt werden.

Bei Heizwiderständen von z.B. einem Elektroboiler gäbe es zwar keinen Anlaufstrom, dennoch würden diese zu Beginn im kalten Zustand aufgrund der Temperaturabhängigkeit des Widerstandes eine andere Leistungsaufnahme aufweisen als im warmen Betriebszustand.

Da die Leistungsspitze bei Kühlschränken relativ kurz ist und sich der Effekt bei vielen unabhängigen Verbrauchern verteilt, wurde wie in [8] und [14] daher auch im Rahmen dieser Arbeit auf die Modellierung der Leistungsspitzen verzichtet.

2.2.4. Scheduled_Operation

In den Geräteklassen `Cont_Operation` und `Thermal_Storage` wurden bisher Gerätetypen modelliert, die ein von der Tageszeit unabhängiges Verhalten aufweisen.

In der Klasse `Scheduled_Operation` und deren Subklassen werden nun alle jene Gerätetypen aufgenommen, auf die das nicht zutrifft. Der Einsatz der Geräte dieser Klasse wird durch Start- und Laufzeiten oder individuelle Lastgänge charakterisiert.

Um die Ermittlung der aktuell vom Gerät aufgenommenen Leistung möglichst effizient in MATLAB umsetzen zu können, wird in dieser Geräteklasse vor dem Simulationsdurchlauf aus den übergebenen Parametern ein sogenannter Einsatzplan erstellt, der definiert, wann das Gerät welche Leistung aufnimmt. Weiters kann in dieser Geräteklasse ein Stand-by-Verbrauch modelliert werden. `Scheduled_Operation` liefert alle Funktionen, die für den Umgang mit einem Einsatzplan notwendig sind für ihre Subklassen `Periodic_Operation`, `Probable_Operation` und `Loadcurve_Operation`. Ähnlich wie die Superklasse

Device kann sie jedoch nicht als Instanz eines Gerätes benutzt werden, da die eigentliche Definition des Einsatzplanes in den jeweiligen Subklassen erfolgt.

2.2.4.1. Einsatzplan

In Tabelle 2.6 sind alle Parameter und Attribute der Geräteklasse `Scheduled_Operation` zusammengefasst. Diese werden auch in den Subklassen benötigt.

Bei der Erzeugung der Geräteinstanzen wird aus den übergebenen Parametern ein Einsatzplan erzeugt. Wie dieser im Detail zu Stande kommt, wird in den Abschnitten 2.2.5.1, 2.2.6.2 und 2.2.7.2 der Subklassen genauer erläutert.

	Name	Einheit	Beschreibung
Parameter	Power_Nominal	W	Anschlussleistung des Geräts
	Power_Stand_by	W	Stand-by-Verbrauch des Gerätes
	Time_Start_Day	-	Liste mit Einschaltzeiten des Gerätes. Die einzelnen Zeitpunkte werden als String der Form 'HH:MM' übergeben (z.B. '12:31')
	Start_Probability	%	Liste mit der zu den Startzeitpunkten gehörenden Einschaltwahrscheinlichkeit zum jeweiligen Startzeitpunkt
	Time_typ_Run	min	Liste mit den zu den Startzeitpunkten gehörenden üblichen Laufzeit des Gerätes zum jeweiligen Startzeitpunkt
Attribute	Activity	-	Ist das Gerät irgendwann im Einsatz? (Nach Erzeugen der Geräteinstanzen können so alle nichtaktiven Geräte aussortiert werden. Daher sollte bei den im Simulationsprogramm verwendeten Geräteinstanzen immer <code>Activity=1</code> sein!)
	Operating	-	Gibt an, ob das Gerät gerade aktiv ist (d.h. eingeschaltet)
	DSM	-	Instanz der Klasse <code>DSM_Device</code> , welche das DSM-Verhalten des Verbrauchers beinhaltet und steuert (siehe Abschnitt 2.3)
	Time_Schedule_Day	-	Fahrplan des Gerätes in laufenden Minuten eines Tages in der Form: [Startzeit, Endzeit, Leistung]
	Time_Start	-	Liste mit den Einschaltzeiten des Gerätes in serieller MATLAB-Zeit
	Time_Schedule	-	Einsatzplan des Gerätes in serieller MATLAB-Zeit
Ausgabe	Power_Input	W	Leistungsaufnahme des Geräts zum aktuellen Zeitpunkt.

Tabelle 2.6: Übersicht der Properties der Klasse `Scheduled_Operation`

Der in den Subklassen generierte Einsatzplan (`Time_Schedule_Day`) ist eine $[m, 3]$ -Matrix. In der ersten Spalte ist die Start-, in der zweiten die Stoppzeit und in der dritten die in diesem Zeitraum aufgenommene Leistung angegeben. In der implementierten Version des Simulationsprogrammes können nur Tageslastgänge definiert werden. Daher werden die Zeiten in laufenden Minuten eines Tages (0 bis 1440 entspricht 0:00 bis 24:00 Uhr) angegeben. Durch Bruchteile von Minuten können beliebig genaue Zeitpunkte definiert werden.

So könnte z.B. ein Einsatzplan eines Gerätes aussehen, das in der Zeit von 11:31:45,2 bis 14:15:56,8 Uhr eine Leistung von 156 W aufnimmt und in der Zeit von 22:50 bis 0:45 Uhr eine Leistung von 250 W:

```
[ 691.753333 855.946667 156.0 ;
 1370.000000 45.000000 250.0 ]
```

Um diesen Einsatzplan für die Simulation verwenden zu können, muss dieser Einsatzplan an die Simulationseinstellungen angepasst und die Zeiten in serielle MATLAB-Zeit-Zahlen (im Folgenden kurz als „MATLAB-Zeit“ bezeichnet) umgerechnet werden. Diese Form der Zeitangabe hat den Vorteil, dass sich mit Hilfe einer einzigen Zahl Datum und Uhrzeit angeben lassen. Weiters können die umfangreichen Funktionen, die für diese Zeitangabe zur Verfügung stehen, verwendet werden. Die MATLAB-Zeit repräsentiert einen Zeitpunkt als Anzahl an Tagen die seit dem „0. Jänner 0000“ vergangen sind. So entspricht der Zeitpunkt „31. Oktober 2010, 12:31“ der MATLAB-Zeit-Nummer 734442,5215277778.

Bevor die Simulation gestartet wird, werden die Zeiten des Einsatzplans von laufenden Minuten in Bruchteile von Tagen umgerechnet. Dies erfolgt in der Methode `adapt_for_simulation` dieser Geräteklasse. Zu diesen Werten wird dann der Startzeitpunkt für die Simulation (der bereits in MATLAB-Zeit vorliegt) addiert, hier zum Beispiel 19.2.2010 0:00 Uhr (erste vertikale Gitterlinie in Diagramm 2.8). Damit ergibt sich ein Einsatzplan wie er in Diagramm 2.8 als blaue Kurve dargestellt ist.

An diesem Verlauf fallen zwei Unstimmigkeiten auf: Zum einen sollte sich der Einsatzplan wiederholen, wenn über mehrere Tage simuliert wird. Zum anderen müsste bereits im Zeitraum von 0:00 bis 0:45 das Gerät aktiv sein.

Damit der Tageslastgang bereits um 0:00 die richtigen Werte aufweist, müssen alle Zeiträume, in denen die Aktivität über Mitternacht hinausgeht, gesondert behandelt werden. Diese Bereiche sind leicht zu identifizieren, da hier $\text{Startzeit} > \text{Endzeit}$ gilt (normalerweise ist $\text{Startzeit} < \text{Endzeit}$). Die Startzeit des betroffenen Bereichs wird um 24 Stunden nach vor verlegt, was zu einem negativen Wert dieses Zeitpunktes führt (angegeben in Tagesbruchteilen):

```
[ 0.480384259 0.594407407 156.0 ;
  -0.048611111 0.031250000 250.0 ]
```

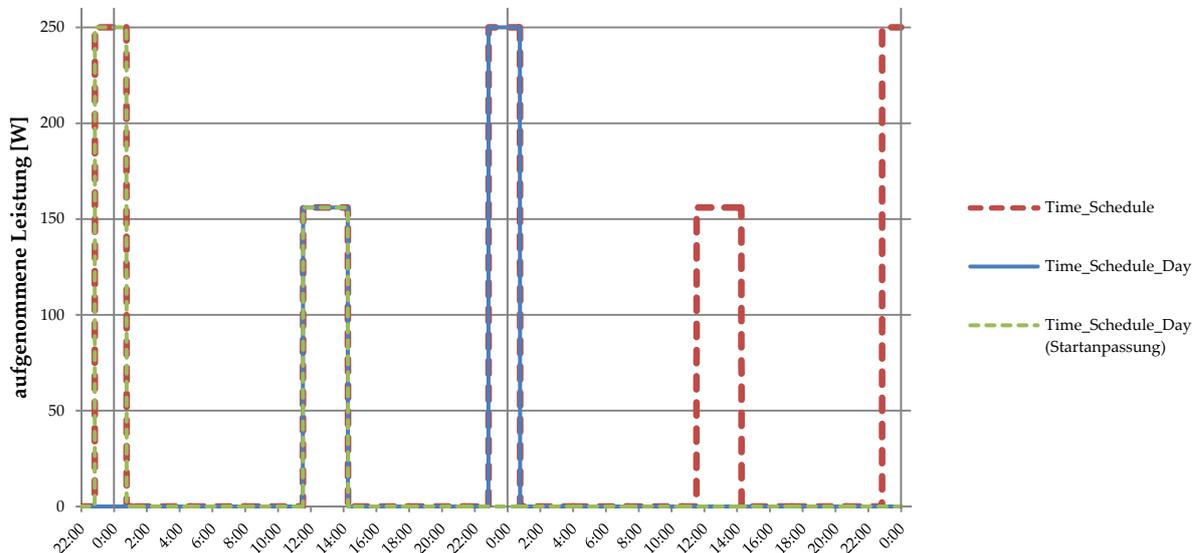


Diagramm 2.8: Illustrierung zum Beispiel für die Umwandlung eines Einsatzplanes

Wird dieser Wert zur Startzeit addiert, ergibt sich für das Beispiel die grüne Kurve in Diagramm 2.8. Zum Startzeitpunkt ist dann das Gerät, wie gefordert, bereits in Betrieb. Um mehrere aufeinanderfolgende Tage zu simulieren (hier im Beispiel insgesamt 48 Stunden) muss nun dieser Einsatzplan dementsprechend oft wiederholt werden. Das fertige Ergebnis entspricht dann der roten Kurve in Diagramm 2.8. Der Einsatzplan für die Simulation `Time_Schedule` (mit Zeiten in MATLAB-Zeit) sieht nun folgendermaßen aus (der 19.02.2010 entspricht 734188):

```
[ 734187.951389 734188.031250 250.0 ;
  734188.480384 734188.594410 156.0 ;
  734188.951389 734189.031250 250.0 ;
  734189.480384 734189.594410 156.0 ;
  734189.951389 734190.031250 250.0 ]
```

Zum Schluss werden alle nicht für die Simulation benötigten Einträge aus dem Einsatzplan entfernt, um später Rechenzeit einzusparen. Das sind jene Einträge, die im angegebenen Simulationszeitraum niemals benötigt werden. Würde sich z.B. der Simulationszeitraum von 19.02.2010 6:00 bis 20.02.2010 16:00 erstrecken, würden aus dem obigen Einsatzplan die erste und die letzte Zeile entfernt.

2.2.4.2. Ermittlung der aufgenommenen Leistung

Mit dem Einsatzplan ist die Leistungsaufnahme des Gerätes zu jedem Zeitpunkt genau bestimmt. Zwar kann nur eine konstante Leistung zwischen zwei Zeitpunkten definiert werden, durch eine entsprechend feine Unterteilung können aber beliebige Verläufe nachgebildet werden, was z.B. bei der Klasse `Loadcurve_Operation` angewendet wird.

Dadurch, dass die Leistungsaufnahme des Gerätes in Form einer Matrix vorliegt, kann der Vorteil von MATLAB bei der Matrizenverarbeitung ausgenutzt werden, wodurch die Rechengeschwindigkeit der Simulation erhöht wird.

Während der Simulation ermittelt das Programm jene Zeitpunkte in MATLAB-Zeit, zu dem die aktuelle Leistung ermittelt werden soll. Je nach Simulationsauflösung liegen die aufeinanderfolgenden Zeitpunkte näher oder weiter auseinander. Bei jedem einzelnen der Zeitpunkte (hier mit `time` bezeichnet) wird nun überprüft, ob er zwischen einer Start- und Endzeit des Einsatzplanes zu liegen kommt. In MATLAB-Code sieht das folgendermaßen aus (`sched = Time_Schedule`, `power = aufgenommene Leistung`):

```
power = sched(sched(:,1)<=time & sched(:,2)>time,3);
```

Trifft dies zu, wird die aktuelle Leistungsaufnahme gleich der Leistung des entsprechenden Zeitraumes gesetzt. Weiters wird das Attribut `Operating` auf 1 gesetzt, da das Gerät zum aktuellen Zeitpunkt aktiv ist. So liefert bei dem obigen Beispiel ein Wert von `time = 734188.5` (entspricht 19.02.2010 12:00 Uhr) `power = 156.0`

Wird kein Zeitraum gefunden, in dem das Gerät aktiv ist, ist `power` ein leeres Array. Dann ist keine Aktivität vorhanden und das Attribut `Operation` wird auf 0 gesetzt. Im Beispiel würde bei einem Wert von `time = 734188.75` (entspricht 19.02.2010 18:00 Uhr) `power = []` und damit das Geräte inaktiv sein.

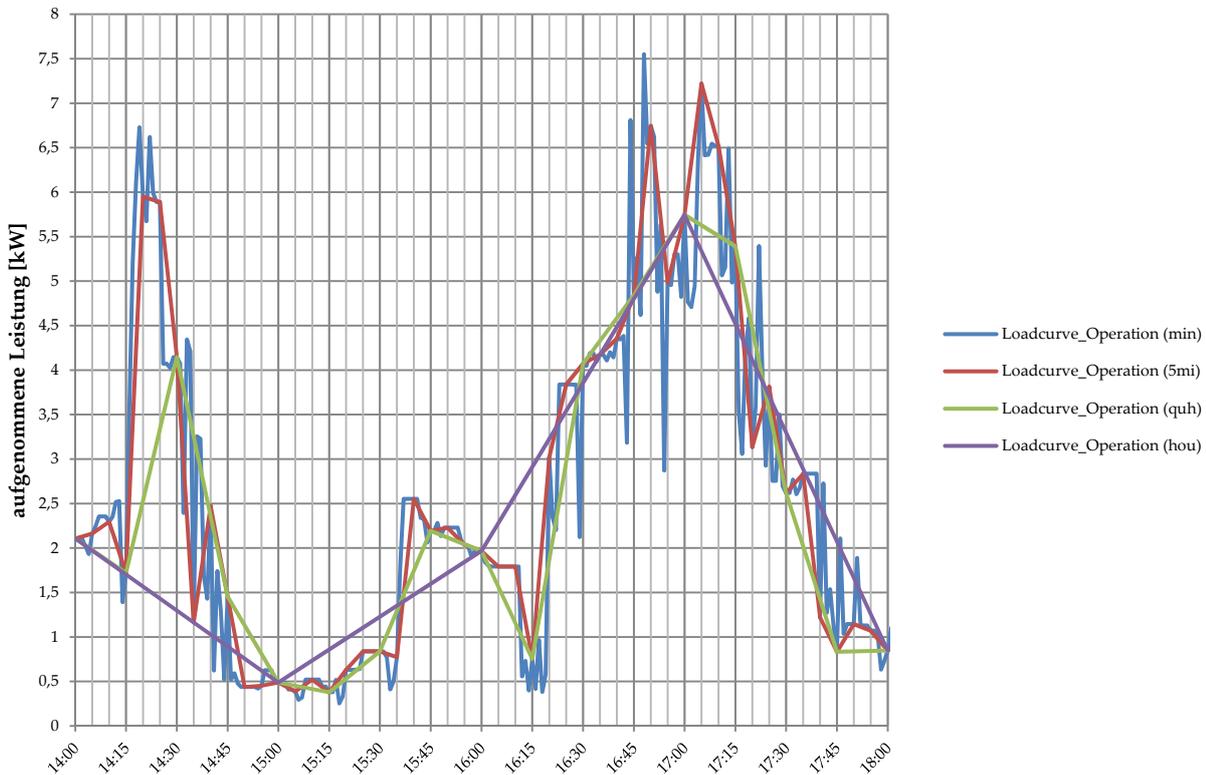


Diagramm 2.9: Vergleich des Simulationsergebnisses der gleichen Geräteinstanzen bei verschiedenen Auflösungen

In Diagramm 2.9 wird anhand eines Beispiels mit ca. 130 Instanzen der Klasse `Loadcurve_Operation` gezeigt, dass unabhängig von der Simulationsauflösung die gleiche aufgenommene Leistung zustande kommt. Darin wurde die Leistung jede Minute (blaue Kurve), alle 5 Minuten (rote Kurve), jede Viertelstunde (grüne Kurve) und jede Stunde ermittelt (violette Kurve). Deutlich ist zu erkennen, dass sich alle Kurven zur vollen Stunde schneiden, welche die grösste Auflösung darstellt. Die ermittelten Werte stellen also unabhängig von der Auflösung die aktuell aufgenommene Leistung zu den Beobachtungszeitpunkten dar.

2.2.4.1. Stand-by-Verbrauch

Mit Hilfe des Parameters `Power_Stand_by` kann ein Stand-by-Verbrauch definiert werden. Dieser tritt nur dann auf, wenn das Gerät nicht aktiv ist (das heißt das Attribut `Operating = 0` ist). So wird der Stand-by-Verbrauch eines realen Gerätes gut angenähert.

In Diagramm 2.10 wird das dargestellt: Der Verbrauch zu den Zeiten, in denen das Gerät aktiv ist, ist derselbe wie im Beispiel, gegeben durch Diagramm 2.8. In den Inaktivitätszeiten tritt der Stand-by-Verbrauch auf, im Beispiel ein relativ hoher Wert von 20 W zur besseren Visualisierung.

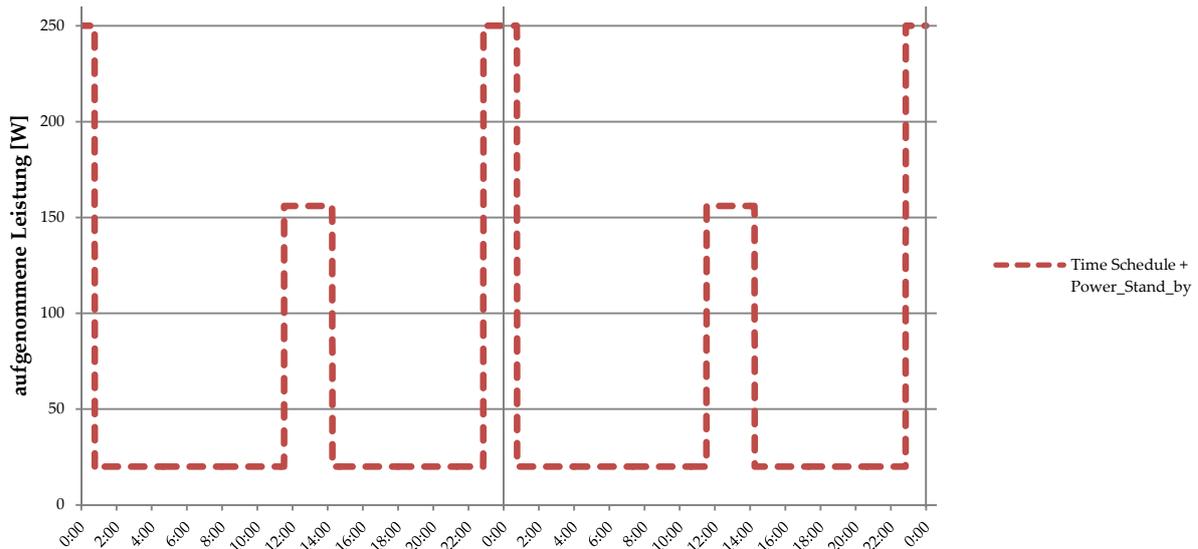


Diagramm 2.10: Auswirkung eines Stand-by-Verbrauchs von 20 W bei dem gleichen Einsatzplan von Diagramm 2.8.

2.2.5. Periodic_Operation

Die Geräteklasse `Periodic_Operation` charakterisiert alle Geräte, die eine sich periodisch wiederholende Leistungsaufnahme aufweisen. Tabelle 2.7 fasst die Parameter und Attribute dieser Klasse zusammen.

	Name	Einheit	Beschreibung
Parameter	Power_Nominal	W	Anschlussleistung des Geräts
	Power_Stand_by	W	Stand-by-Verbrauch des Gerätes
	Time_Start_Day	-	Startzeit für eine Aktivitätszeit des Gerätes. Wird als String der Form 'HH:MM' übergeben (z.B. '12:31')
	Start_Probability	%	Wahrscheinlichkeit, dass das Gerät verwendet wird
	Time_typ_Run	min	Dauer einer Aktivitätsphase des Gerätes
	Time_Period	min	Periodendauer der Aktivität des Gerätes
Attribute	Activity	-	Ist das Gerät irgendwann im Einsatz? (Nach Erzeugen der Geräteinstanzen können so alle nichtaktiven Geräte aussortiert werden. Daher sollte bei den im Simulationsprogramm verwendeten Geräteinstanzen immer <code>Activity = 1</code> sein!)
	Operating	-	Gibt an, ob das Gerät gerade aktiv ist (d.h. eingeschaltet)
	DSM	-	Instanz der Klasse <code>DSM_Device</code> , welche das DSM-Verhalten des Verbrauchers beinhaltet und steuert (siehe Abschnitt 2.3)
	Time_Schedule_Day	-	Fahrplan des Gerätes in laufenden Minuten eines Tages in der Form: [Startzeit, Endzeit, Leistung]
	Time_Start	-	Liste mit den Einschaltzeiten des Gerätes in serieller MATLAB-Zeit
	Time_Schedule	-	Einsatzplan des Gerätes in serieller MATLAB-Zeit
	Ausgabe	Power_Input	W

Tabelle 2.7: Übersicht der Properties der Klasse `Periodic_Operation`

In den für diese Arbeit verwendeten Gerätekonstellationen wird diese Geräteklasse nicht verwendet. Sie diente anfangs dazu, das Verhalten von thermischen Speichern (wie z.B. Kühlschränken), das ja ein periodisches ist, nachzubilden. Der Vorteil lag in der schnelleren Simulation durch den vor der Simulation definierten Einsatzplan. Wenn aber eine DSM-Funktionalität die Geräteeigenschaften steuern will (z.B. die Soll-Temperatur verändern) wäre die Implementierung der Auswirkungen zu aufwändig und mit Unsicherheiten behaftete gewesen, weshalb für diese Gerätetypen dann die Klasse `Thermal_Storage` entwickelt wurde.

Andere Gerätetypen, die diese Klasse repräsentieren könnte, wurden nicht für die Simulation benötigt. Der Vollständigkeit halber werden aber die Eigenschaften dieser Klasse noch genauer erläutert.

2.2.5.1. Erstellung des Einsatzplanes

Die Aktivität eines Gerätes der Klasse `Periodic_Operation` wird definiert durch die Periodendauer `Time_Period` und die Aktivitätszeit `Time_typ_Run`, wobei normalerweise `Time_Period < Time_typ_Run` gilt. Ist das nicht der Fall, ist das Gerät dauerhaft im Einsatz.

Wird zusätzlich eine Startzeit `Time_Start_Day` angegeben, wird dafür gesorgt, dass zu diesem angegebenen Zeitpunkt das Gerät eine Aktivitätszeit startet (siehe Beispiel in Diagramm 2.11). Falls keine Startzeit angegeben wird, wird diese zufällig gemäß einer Gleichverteilung über einen ganzen Tag ermittelt.

2.2.5.2. Beispiel einer Instanz von `Periodic_Operation`

Diagramm 2.11 zeigt das Ergebnis einer Simulation eines Gerätes der Klasse `Periodic_Operation` mit den Parameterwerten aus Tabelle 2.8. Das Gerät ist alle 74 min für 37 min aktiv. Als Startzeit einer Aktivitätsphase wurde 12:00 Uhr angegeben, wodurch die Lage der einzelnen Aktivitätszeiträume bestimmt wird. Alle Werte der Standardabweichung wurden Null gesetzt, um die exakte Widerspiegelung der Parameterwerte im Ergebnis zu ermöglichen.

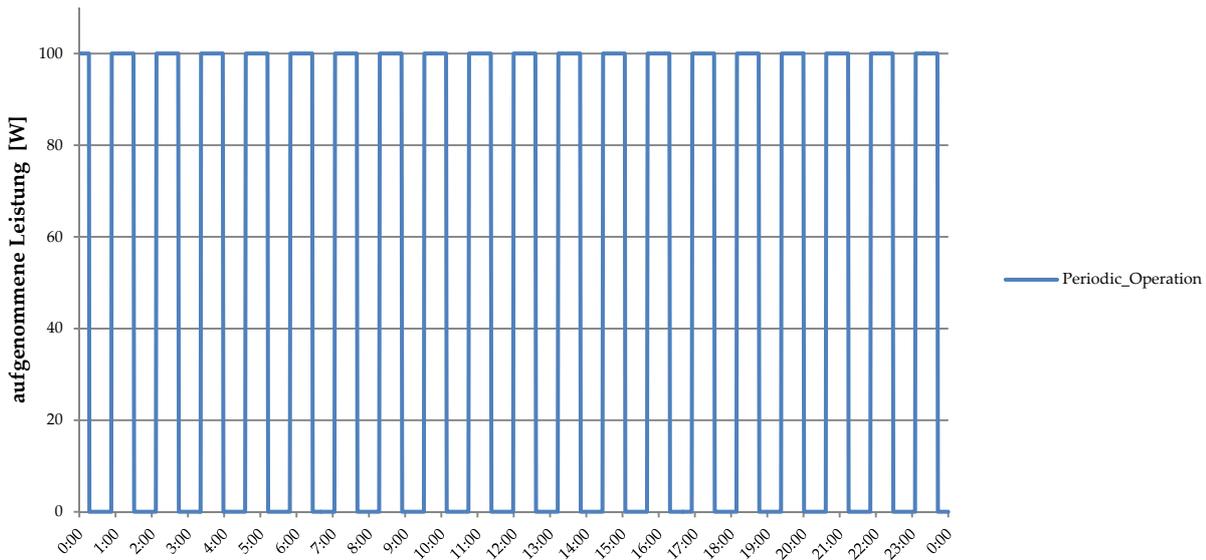


Diagramm 2.11: Beispiel einer Instanz von `Periodic_Operation` mit den Parameterwerten aus Tabelle 2.8

<code>Power_Nominal</code>	100	W	0	%
<code>Time_Start</code>	12:00	Uhr	0	min
<code>Start_Probability</code>	100	%		
<code>Time_typ_Run</code>	37	min	0	%
<code>Time_Period</code>	74	min	0	%

Tabelle 2.8: Parameterwerte für das Beispiel einer Instanz von `Periodic_Operation` (Auszug aus Parameterdatei)

2.2.6. Probable_Operation

In der Geräteklasse `Probable_Operation` werden jene Gerätetypen charakterisiert, deren Aktivitätszeiträume sich zufällig aufgrund von stochastischem Nutzerverhalten ergeben. Während ihrer Nutzung haben diese Geräte eine konstante Leistungsaufnahme. Beispiele für Geräte dieses Typs wären Fernseher oder Beleuchtungskörper.

Jedes Gerät weist ein individuelles Verhalten auf, welches nicht vorhergesagt werden kann. Wird jedoch ein großes Kollektiv dieser Geräte betrachtet, können zumindest Vorhersagen über das Verhalten des Kollektivs gemacht werden, die auch mit großer Wahrscheinlichkeit zutreffen.

Damit nun das Simulationsmodell mit individuell agierenden Geräten ausgestattet werden kann, wird der Einsatz jedes Gerätes aufgrund von Parametern, die das Verhalten des gesamten Kollektivs widerspiegeln, ermittelt. In Summe trägt so jedes individuell arbeitende Gerät zum geforderten Gesamtverlauf bei.

2.2.6.1. Parameter

Die Geräteklasse `Probable_Operation` besitzt die gleichen Parameter und Attribute wie ihre Superklasse `Scheduled_Operation`. Diese sind bereits in Tabelle 2.6 auf Seite 20 aufgelistet.

Diese Klasse wird charakterisiert durch eine Liste mit Einschaltzeitpunkten `Time_Start`, die Breite der Verteilung der einzelnen Startzeiten um den jeweiligen Einschaltzeitpunkt, eine typische Laufzeit, in der

das Gerät ab dem jeweiligen Einschaltzeitpunkt aktiv ist (`Time_typ_Run`) sowie einer Startwahrscheinlichkeit zu diesem Zeitpunkt (`Start_Probability`).

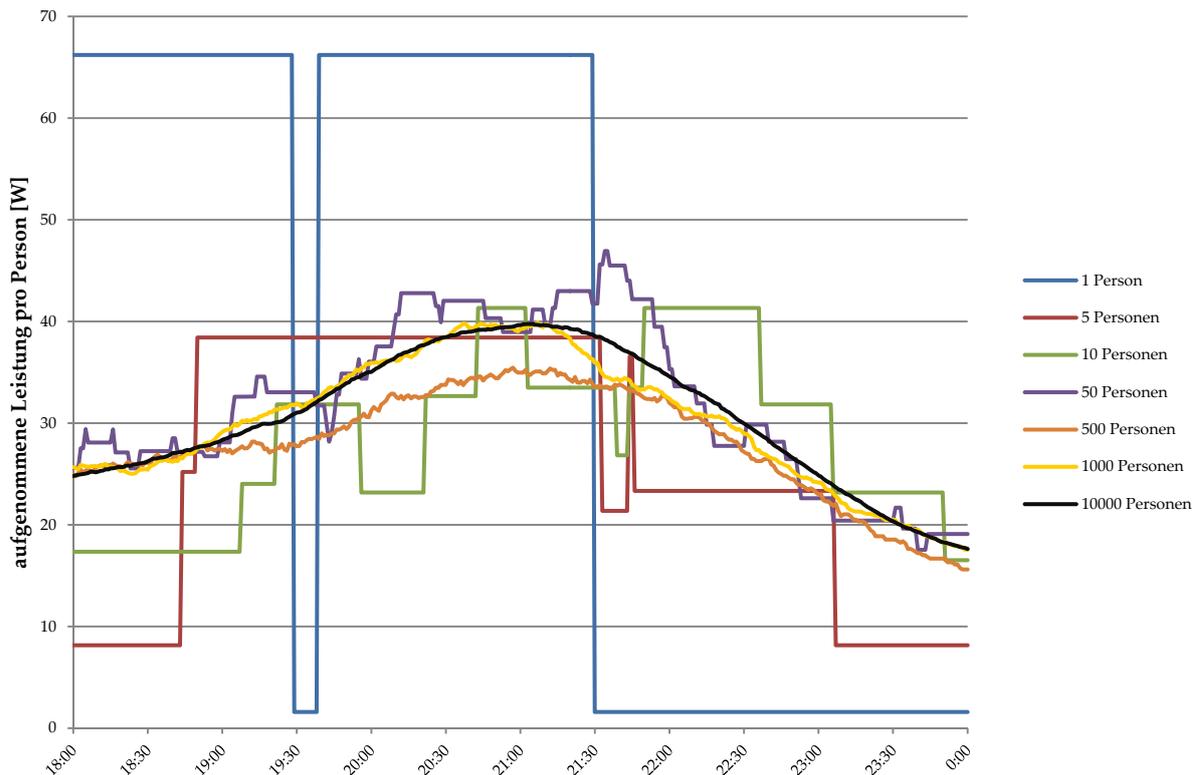


Diagramm 2.12: Zusammenspiel von Einsatzplänen für verschiedene Benutzerzahlen bei der Klasse `Probable_Operation`. Als Parameter wurden für das obige Beispiel die Werte von Tabelle 3.8 auf Seite 64 herangezogen (Audio-Video-Geräte)

Mit Hilfe dieser Parameter kann der Tageslastgang einer Verbrauchergruppe nachgebildet werden, indem die Parameterwerte dementsprechend gelegt werden. In Diagramm 2.12 ist die Entwicklung von einem Einzeleinsatzplan bis zu der Leistungsaufnahme eines großen Gerätekollektivs dargestellt. Je mehr Geräte simuliert werden, desto mehr gleicht der sich ergebende Verlauf an den erwarteten Verlauf an.

2.2.6.2. Erstellung des Einsatzplanes

Bei der Erzeugung der Geräteinstanzen werden die Startzeitpunkte `Time_Start` sowie die dazugehörigen typischen Laufzeiten `Time_typ_Run` gemäß der angegebenen Standardabweichung normalverteilt gestreut. So hat jedes Gerät bereits zum Zeitpunkt der Erstellung des Einsatzplanes individuelle Start- und Laufzeiten.

Mit diesen Angaben wird ein kompletter Einsatzplan erstellt. Daran anschließend wird anhand der Startwahrscheinlichkeit `Start_Probability` zufällig für jeden Eintrag des Einsatzplanes ermittelt, ob dieser erhalten bleibt, oder aber, da das Gerät zu diesem Zeitpunkt doch nicht startet, aus der Liste entfernt wird.

Ist das Geräte zu keinem Zeitpunkt in Betrieb, wird bei der Instanzenerzeugung der Wert von `Activity = 0` gesetzt, wodurch das Geräte bei der späteren Simulation nicht berücksichtigt werden wird.

2.2.6.3. Auswirkungen der Parameter auf das Simulationsergebnis

Für die vorliegende Arbeit wurden für einen typischen Lastgang die einzelnen Parameter per Hand ermittelt. In zukünftigen Versionen des Simulationsprogramms wird die Verwendung von Einschaltwahrscheinlichkeiten zu bestimmten Zeitpunkten angedacht, da diese Daten für viele Gerätegruppen bereits vorhanden sind.

Diagramm 2.13 zeigt, wie es zu der Verteilung des Geräteeinsatzes kommt und wie sich das Variieren der einzelnen Parameter dieser Klasse auswirkt.

Bei der diesem Beispiel zugrundeliegenden fiktiven Gerätkonstellation hat jedes Gerät exakt die Leistungsaufnahme von 1 W. Um drei Zeitpunkte herum werden alle Geräte gestartet. Jedes Gerät wird also während des betrachteten Zeitraums dreimal gestartet und ist dann individuell eine gewisse Zeitspanne aktiv. Wie lange dies ist und wann genau jedes Gerät startet, wird zufällig gemäß Abschnitt 2.2.6.2 ermittelt. Insgesamt wurden hier, um glatte Verläufe zu erhalten, 10.000 Geräte simuliert.

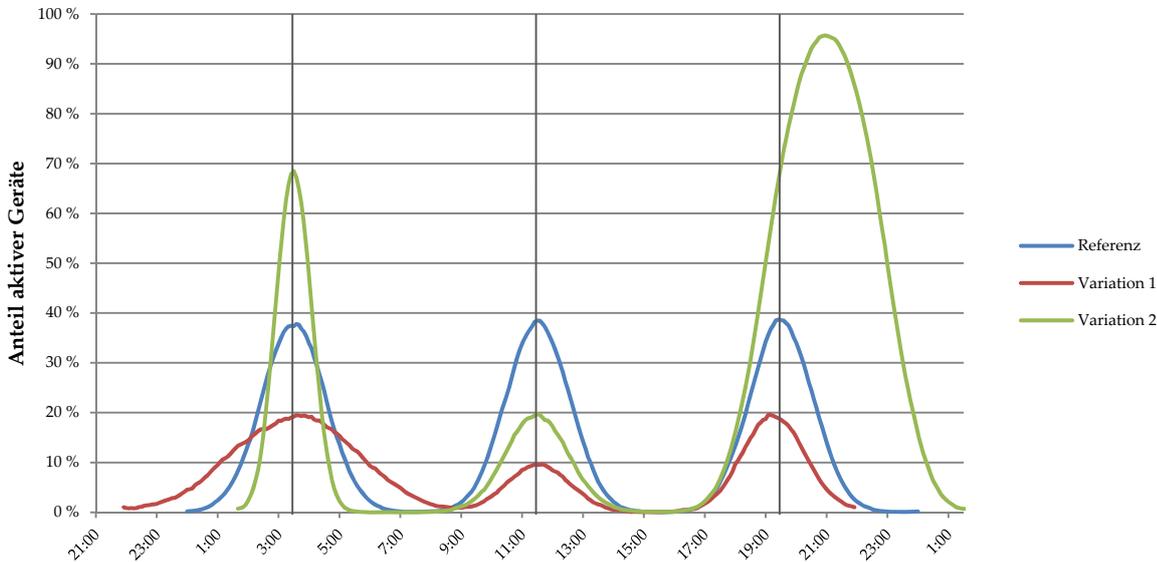


Diagramm 2.13: Darstellung der Auswirkung der Parameter in der Klasse "Probable_Operation"

Die blaue Kurve dient als Referenz. Hierbei wurden die Parameter für jeden Startzeitpunkt gleich belassen. Die folgende Tabelle zeigt einen Auszug aus der Parameterdatei für diese Simulation.

Power_Nominal	1	W	0	%
Power_Stand_by	0	W	0	%
Time_Start	03:00	11:00	19:00	Uhr
'-> Std.Dev.:	60	60	60	min
Start_Probability	100	100	100	%
Time_typ_Run	60	60	60	min
'-> Std.Dev.:	0	%		

Tabelle 2.9: Parameter der Referenzkurve für Diagramm 2.13 (Auszug aus Parameterdatei)

Man erkennt, dass jeweils um 03:00, 11:00 und 19:00 die Startwerte angesiedelt werden. Die Streuung um diesen Startwert ist mit 60 min angegeben, das heißt, dass die realen Startwerte mit einer Standardabweichung der Breite von 60 min um den angegebenen Startwert verteilt sind. In der Zeit von z.B. 01:00 bis 05:00 starten somit 95,4% der Geräte, in der Zeit von 00:00 bis 06:00 haben 99,7 % ihre Startzeit.

Wenn das Gerät gestartet hat, läuft es exakt 60 min lang (Time_typ_Run). Dies führt dazu, dass das Maximum der Anzahl der aktiven Geräte nicht zu dem angegebenen Startzeitpunkt erreicht wird, sondern erst im Schnitt um die Hälfte von Time_typ_Run später (im vorliegenden Beispiel jeweils eine halbe Stunde). Die Zeitpunkte, zu denen das Maximum der aktiven Geräte der Referenzkurve auftreten sollte, sind in Diagramm 2.13 als vertikale Gitternetzlinien markiert.

Variierter Parameter:	03:00		11:00		19:00	
	Time_Start '-> Std.Dev.		Start_Probability		Time_typ_Run	
Referenz	60 min		100 %		60 min	
Variation 1	120 min		50 %		240 min	
Variation 2	30 min		25 %		30 min	

Tabelle 2.10: Übersicht über die Variationen der verschiedenen Parameter für Diagramm 2.13

Tabelle 2.10 gibt eine Übersicht darüber, wie die Parameter jeweils geändert wurden. Pro Uhrzeit wurde immer der gleiche Parameter entweder erhöht oder verringert.

Sehr gut sind die glockenförmigen Kurvenverläufe einer Normalverteilung zu erkennen, was aufgrund der Parametervariierung gemäß einer Normalverteilung nicht überraschend ist.

Um 3:00 Uhr wird die Standardabweichung der Startzeit verändert. Wird diese verdoppelt (rote Kurve) ist die Streuung der Startzeiten doppelt so hoch, wodurch die Glockenkurve der aktiven Geräte doppelt so breit wird. Auch reduziert sich der Anteil der aktiven Geräte im Maximum. Umgekehrt bei einer Halbierung (grüne Kurve): Mehr Geräte starten gleichzeitig, weshalb sich das Maximum erhöht, die Startzeiten rücken näher zusammen. Der Zeitpunkt des Maximums bleibt aber immer gleich. Würde die Standardabweichung zu Null gesetzt, würden alle Geräte gleichzeitig starten und ein rechteckförmiger Verlauf mit 100% aktiven Geräten über 60 min zustande kommen.

Um 11:00 Uhr wird die Einschaltwahrscheinlichkeit halbiert (rote Kurve) bzw. auf 25% (grüne Kurve) gesetzt. Diese Variierung beeinflusst nicht die Breite der Glockenkurve sondern nur deren Höhe. Gut ist die verringerte Aktivität der Geräte im Vergleich mit der Referenzkurve zu erkennen.

Um 19:00 Uhr werden unterschiedliche, typische Laufzeiten definiert. Laufen alle Geräte im Schnitt um die Hälfte kürzer (grüne Kurve) kommt es einerseits zu einer Verschmälerung und Reduzierung des Maximums der Glockenkurve. Dies kommt dadurch zu Stande, da aufgrund der verringerten Laufzeit nicht mehr so viele Geräte zu den betrachteten Zeitpunkten gleichzeitig im Einsatz sind. Andererseits verschiebt sich das Maximum nach links (von 19:30 auf 19:15 Uhr). Genau umgekehrt ist das Bild bei einer Vervierfachung der Laufzeit: Das Maximum wird um 21:00 Uhr erreicht und beinahe alle Geräte sind zu diesem Zeitpunkt im Einsatz. Würde die Laufzeit noch mehr erhöht, würde die Glockenkurve bei 100% aktive Geräte abflachen.

Die Verschiebung des Maximums bei Änderung der typischen Laufzeit stellte sich bei der Parameterfindung als umständlich heraus, da immer `Time_Start` und `Time_typ_Run` verändert werden mussten, damit das Maximum der aktiven Geräte an gleicher Stelle bleibt. Für eine künftige einfachere Parameterfindung könnte von einer Angabe einer Startzeit abgegangen und ein Zeitpunkt maximaler Aktivität definiert werden. Die Startzeit würde sich dann aus dem Zeitpunkt der maximaler Aktivität reduziert um die halbe Laufzeit zu diesem Zeitpunkt ergeben, wodurch nur mehr die Breite und Höhe des Anteils aktiver Geräte beeinflusst würde. Da aber bereits für die Simulationen, die in dieser Arbeit durchgeführt wurden, alle Parameter vorhanden sind, wurde diese Änderung nicht mehr implementiert.

2.2.7. Loadcurve_Operation

Die Klasse `Loadcurve_Operation` ist eine Subklasse von `Probable_Operation`. Im Gegensatz zu den Geräten, die diese Klasse repräsentiert und die eine konstante Eingangsleistung aufweisen, charakterisiert `Loadcurve_Operation` jene Geräte, deren Einsatz zwar auch statistisch verteilt ist, deren Leistungsaufnahme aber nach einem definierten Ablauf (Programm, Lastkurve) erfolgt.

2.2.7.1. Parameter

In Tabelle 2.12 sind die Parameter und Attribute der Klasse `Loadcurve_Operation` zusammengefasst. Sie sind, bis auf die Angaben zur Lastkurve des Gerätes, gleich wie bei der Superklasse `Probable_Operation`. So werden genauso Startzeiten (`Time_Start_Day`) und Startwahrscheinlichkeiten zu diesen Startzeiten (`Start_Probability`) definiert, die den Geräteeinsatz beeinflussen.

Die Lastkurve `Power_Loadcurve` definiert einen Ablauf der Leistungsaufnahme als eine Zwei-Spalten-Matrix. In der ersten Spalte wird die Dauer, in der zweiten Spalte die Höhe der Leistungsaufnahme im aktuellen Aktivitätsbereich angegeben.

Zum Beispiel könnte eine Lastkurve folgendermaßen aussehen:

Power_Loadcurve	min	W	
	10	200	
	20	100	
	30	50	
	40	25	
	30	50	
	20	100	
	10	200	
'-> Std.Dev.:	10	20	%

Tabelle 2.11: Beispiel für eine Lastkurve

Darin wird die ersten 10 Minuten eine Leistung von 200 W aufgenommen, dann 20 Minuten 100 W usw. Die Standardabweichung kann für die Spalte der Dauer und der Leistung unterschiedlich definiert werden, um so unterschiedliche Programme zu simulieren (z.B. unterschiedliche Beladung und Programmauswahl bei einer Waschmaschine).

	Name	Einheit	Beschreibung
Parameter	Power_Nominal	W	Anschlussleistung des Gerätes
	Power_Stand_by	W	Stand-by-Verbrauch des Gerätes
	Time_Start_Day	-	Startzeit für eine Aktivitätszeit des Gerätes. Wird als String der Form 'HH:MM' übergeben (z.B. '12:31')
	Start_Probability	%	Wahrscheinlichkeit, dass das Gerät verwendet wird
	Power_Loadcurve	-	Lastkurve des Gerätes ([m,2]-Matrix): [Dauer (in min), Leistung (in W)] Diese wird Zeile für Zeile abgearbeitet. Es können aber auch n Lastkurven vorliegen ([m, 2n]-Matrix), es wird dann eine Lastkurve pro Startzeitpunkt ausgewählt (gemäß Loadcurve_Allocation). Dieser Parameter wird während Objekterzeugung in eine Struktur mit den einzelnen Lastkurven umgewandelt (siehe Funktion CALCULATE_SCHEDULE).
	Loadcurve_Allocation	-	Wahrscheinlichkeitsdichte für die einzelnen Lastkurven, falls mehrere angegeben wurden. Mit dieser Angabe wird ermittelt, welche Lastkurve zu welchem Startzeitpunkt zur Anwendung kommt. Bei n Lastkurven eine [1,n]-Matrix
Attribute	Loadcurve_non_stop_Parts	-	[m, 2] Indexliste jener Teile der Lastkurve bzw. der Lastkurven (dann [m, 2n] Matrix), die nicht unterbrochen werden dürfen, in der Form: [Start_Index, End_Index]
	Activity	-	Ist das Gerät irgendwann im Einsatz? (Nach Erzeugen der Geräteinstanzen können so alle nichtaktiven Geräte aussortiert werden. Daher sollte bei den im Simulationsprogramm verwendeten Geräteinstanzen immer Activity = 1 sein!)
	Operating	-	Gibt an, ob das Gerät gerade aktiv ist (d.h. eingeschaltet)
	DSM	-	Instanz der Klasse DSM_Device, welche das DSM-Verhalten des Verbrauchers beinhaltet und steuert (siehe Abschnitt 2.3)
	Time_Schedule_Day	-	Fahrplan des Gerätes in laufenden Minuten eines Tages in der Form: [Startzeit, Endzeit, Leistung]
	Time_Start	-	Liste mit den Einschaltzeiten des Gerätes in MATLAB-Zeit
	Time_Schedule	-	Einsatzplan des Gerätes in serieller MATLAB-Zeit und angepasst an Simulationseinstellungen
	Time_Stop_Day	min	Endzeiten der einzelnen Lastgänge in laufenden Minuten eines Tages
	Time_Stop	-	Endzeiten der einzelnen Lastgänge in MATLAB-Zeit
Picked_Loadcurves	-	gibt an, welche Lastkurve zu welcher Startzeit gewählt wurde	
Ausgabe	Power_Input	W	Leistungsaufnahme des Gerätes zum aktuellen Zeitpunkt.

Tabelle 2.12: Übersicht der Properties der Klasse Loadcurve_Operation

Es besteht aber auch die Möglichkeit, mehrere verschiedene Lastkurven für ein Gerät anzugeben. Diese müssen in keiner Weise in der Anzahl der Schritte bzw. in der Dauer und Leistungsaufnahme gleich sein, sondern können beliebig definiert werden, wie z.B. in Tabelle 2.13 gezeigt wird.

Power_Loadcurve	min	W	min	W	
	10	200	6	20	
	15	100	12	40	
	20	50	18	80	
	25	25	24	160	
	20	50	30	320	
	15	100	24	160	
	10	200	18	80	
			12	40	
			6	20	
'-> Std.Dev.:	10	20	20	30	%
Loadcurve_Allocation	30	70	%		
Loadcurve_non_stop_Parts	Start_idx	End_idx	Start_idx	End_idx	
	1	3	3	7	
	5	7			

Tabelle 2.13: Beispiel für die Angabe mehrerer Lastkurven mit deren Auftritts-Häufigkeit sowie der Definition von Programmteilen, die im laufenden Betrieb nicht unterbrochen werden sollen

Die Standardabweichung kann hier ebenfalls für jede einzelne Lastkurve separat angegeben werden oder nur einmal für alle Lastkurven (Wegfall der roten Werte in Tabelle 2.13).

Der Parameter `Loadcurve_Allocation` bestimmt, mit welcher Wahrscheinlichkeit die jeweiligen Lastkurven auftreten. Im Beispiel wird in 30% der Einsätze die erste, in 70% die zweite Lastkurve herangezogen.

Eine weitere Eigenschaft, die erst bei Einsatz einer DSM-Funktion zu tragen kommen wird, ist die Definition jener Teile einer Lastkurve, die während des Betriebes immer hintereinander ohne Unterbrechung durchgeführt werden müssen. Dies könnte z.B. bei einer intelligenten Waschmaschine der Fall sein, in deren Programmablauf ein DSM-Chip eingreifen kann und die jene Programmteile, deren Unterbrechung einen Nachteil beim Waschvorgang bringen (z.B. dass das warme Waschwasser aufgrund einer Deaktivierung des Gerätes abkühlt und dann wieder aufgeheizt werden muss oder die Wäsche nicht dauerhaft in der Lauge liegen sollte) zuerst komplett absolviert. Näheres dazu im Abschnitt 2.3.2.6. Da nur das Gerät selbst wissen kann, welche Programmteile nicht unterbrochen werden dürfen, müssen diese auch als Geräteeigenschaft definiert werden.

Bei `Loadcurve_non_stop_Parts` erfolgt die Angabe der nicht unterbrechbaren Programmteile anhand der Indizes der jeweiligen Lastkurve. Dies erfolgt wieder in einer Zwei-Spalten Matrix für jede Lastkurve, wobei in der ersten Spalte der Start-, in der zweiten der End-Index des Bereichs, der nicht ununterbrochen werden darf, angegeben wird. Die angegebenen Indizes werden als zu diesem Bereich gehörend betrachtet. Im Beispiel von Tabelle 2.13 werden Programmteile als nicht unterbrechbar definiert, die die drei höchsten Leistungswerte aufweisen. Das heißt konkret, dass z.B. bei der ersten Lastkurve die ersten und letzten 45 Minuten nicht unterbrochen werden dürfen.

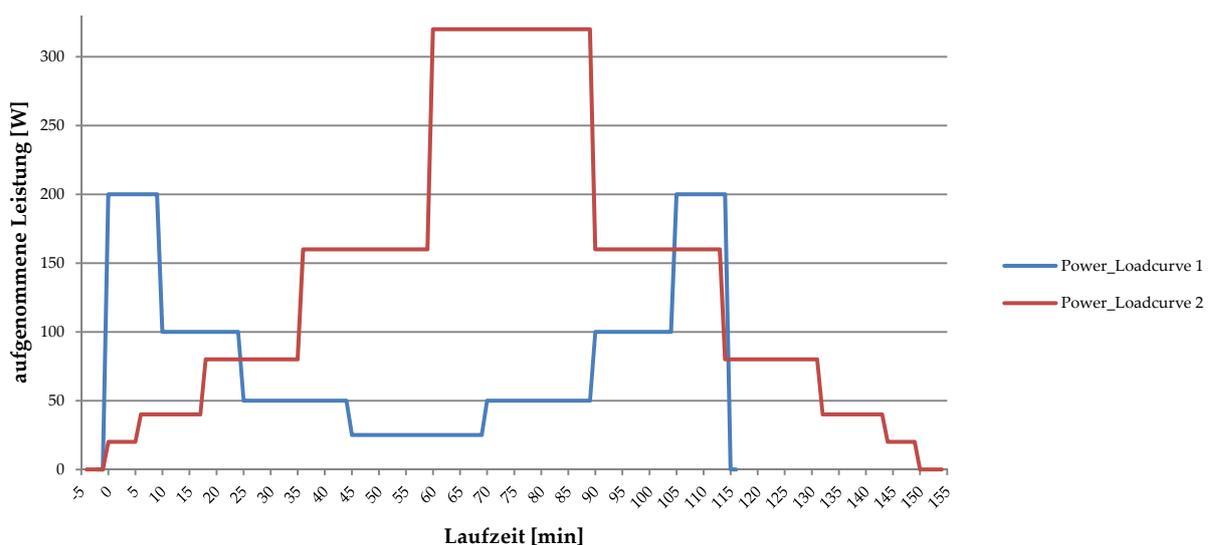


Diagramm 2.14: Darstellung der Lastkurven aus dem Beispiel angegeben in Tabelle 2.13

2.2.7.2. Erstellung des Einsatzplanes

Beim Erstellen des Einsatzplanes wird zuerst ermittelt, zu welchen (bei der Instanzenerzeugung bereits gestreuten) Startzeiten das Gerät gemäß der angegebenen Startwahrscheinlichkeit aktiv wird. Falls mehrere verschiedene Lastkurven angegeben wurden, wird für jede Startzeit gemäß der Verteilung `Loadcurve_Allocation` eine Lastkurve ausgewählt und diese Auswahl gespeichert.

Schließlich wird aus den ausgewählten Lastkurven ein Einsatzplan erstellt, indem bei jedem Startzeitpunkt die zugeordnete Lastkurve eingefügt wird. Dabei wird überprüft, ob es eventuell zu Überschneidungen von verschiedenen Lastkurven kommt, weil z.B. die Startzeit der nächsten Lastkurve in die Laufzeit der vorhergehenden Lastkurve fällt. Dazu wird jener Zeitpunkt ermittelt, zu dem die aktive Lastkurve beendet ist. Ist nun der Startzeitpunkt der nächsten Lastkurve vor dem Endzeitpunkt der vorhergehenden, wird die nächste Lastkurve erst zum Endzeitpunkt der vorhergehenden Lastkurve gestartet.

Diagramm 2.15 zeigt dies anhand von den Lastkurven, die in Tabelle 2.13 angegeben wurden. In diesem Beispiel wurden die Startzeiten 12:00 und 13:00 Uhr mit jeweils 100% Startwahrscheinlichkeit vorgegeben (vertikale Gitternetzlinien in Diagramm 2.15). Beide Lastkurven haben eine Wahrscheinlichkeit von 50%.

Ausgewählt wurde ein Simulationsdurchlauf, in dem zuerst die erste und dann die zweite Lastkurve gestartet werden. Es ist gut zu erkennen, dass zuerst die erste Lastkurve vollständig absolviert wird und dann erst die zweite unmittelbar danach startet.

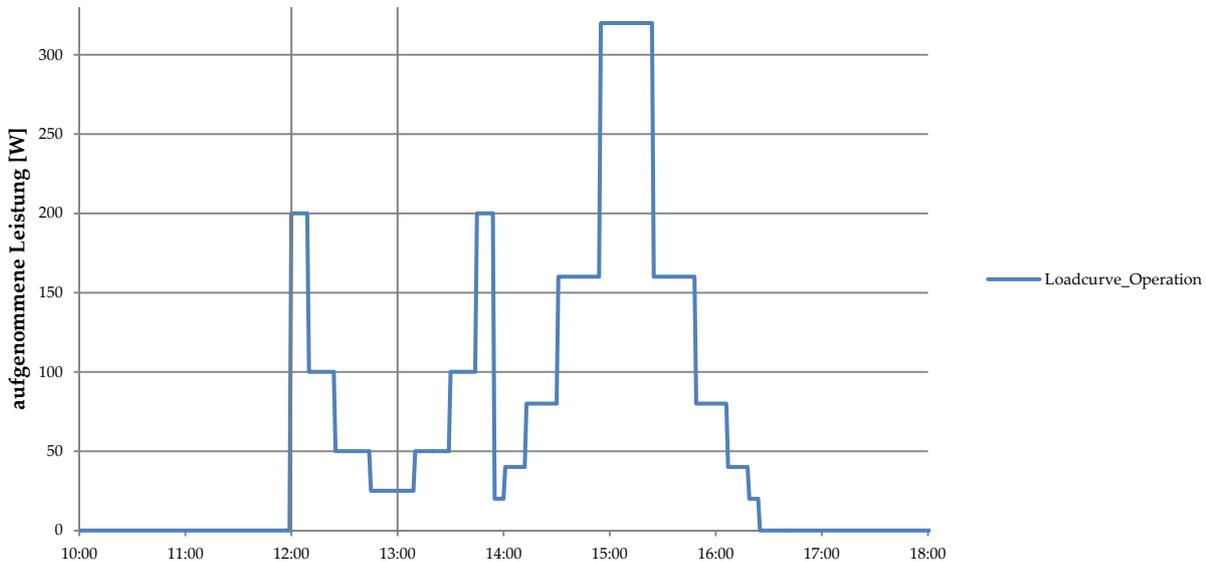


Diagramm 2.15: Verschiebung der Lastkurven bei kurz aufeinander folgenden Startzeitpunkten mit Lastkurven angegeben in Tabelle 2.13

Dies setzt jedoch voraus, dass der Benutzer des Gerätes dieses, sobald es sein Programm beendet hat, es wieder in einen betriebsbereiten Zustand versetzt. So müsste er z.B. nachdem die Waschmaschine mit einem Waschgang fertig ist, diese wieder mit Wäsche beladen und starten. Das muss, wenn man Diagramm 2.15 betrachtet, auch noch in sehr kurzer Zeit erfolgen. Das sind natürlich eher unwahrscheinliche Annahmen.

Um die Zeit zu berücksichtigen, die ein Benutzer benötigt, um z.B. eine Waschmaschine auszuräumen und wieder mit Wäsche zu beladen, könnte bei jeder Lastkurve am Ende von dieser eine Zeitspanne mit geringer bzw. ohne Leistungsaufnahme definiert werden (wie es z.B. in Tabelle 3.19 für die Waschmaschinen gemacht wurde). Dennoch bleibt immer noch die Annahme, dass ein Benutzer da ist, der dies auch immer zeitgerecht durchführt. Geht man aber davon aus, dass die Waschmaschine unbeobachtet arbeiten wird, insbesondere wenn deren Start- und Laufzeit aufgrund des Eingriffs durch DSM nicht genau abgeschätzt werden kann (siehe Abschnitt 2.3.2.6), ergeben sich dann zwangsläufig lange Standzeiten zwischen den einzelnen Startzeiten.

Da aber der Fall, dass mehrere Programme am selben Tag gestartet werden, eher selten vorkommt, wird auf die eben beschriebenen Umstände keine weitere Rücksicht genommen.

2.3. Simulation der DSM-Funktionalität

Die in dieser Arbeit untersuchte Form von DSM soll vordergründig das Ziel verfolgen, die Geräte dahingehend zu beeinflussen, dass sie aktiv für den Erhalt des funktionstüchtigen Netzzustandes arbeiten. Sie sollen sich kooperativ gegenüber dem Versorgungsnetz und damit auch den Energieversorgern verhalten, um Mangelsituationen entschärfen zu können.

Um Verbraucher kooperativ aus Netzsicht zu machen, müssen diese in irgendeiner Weise mit einer zusätzlichen Einrichtung versehen werden, damit sie auf die aktuelle Netzsituation reagieren können. In dieser Arbeit wird diese Einrichtung als eigenständiger „Chip“ angesehen, der im Gerät verbaut ist, aber nicht notwendigerweise vollkommen in dieses integriert werden muss. Vielmehr stellt dieser Chip eine Schnittstelle zur Verfügung, über die die Informationen zur DSM-Funktionalität an das Gerät übermittelt werden. Diese Information besteht nun vordergründig darin, ob ein Eingriff in das Geräteverhalten erfolgen soll oder nicht. Die Reaktion des Gerätes erfolgt dann, je nach Gerätetyp, mit der vom Gerät ermöglichten Art und Weise.

Im einfachsten Fall könnte der Chip ein Relais betätigen, der das Gerät einfach vom Netz nimmt. Bei einer höheren Integration könnte die vom Chip generierte Information von der Gerätesteuerung aufgegriffen werden und die Steuerung dann in das Geräteverhalten eingreifen. Wichtig ist aber hier der Ansatz, dass der Chip nichts von den Eigenschaften des Gerätes weiß, also ob es sich nun z.B. um einen Fernseher oder um einen Kühlschrank handelt. Jeder Chip ist gleich und stellt lediglich die Information zur Verfügung, ob ein Reagieren notwendig ist, alle weiteren Handlungsmaßnahmen finden auf der Geräteseite statt.

Dieser universelle Chip hätte den Vorteil, dass er in hoher Stückzahl kostengünstig produziert werden könnte. Mit Hilfe einer Standardisierung der Schnittstelle, könnte jeder Gerätehersteller selbst entscheiden, inwieweit die Funktionen in sein Gerät integriert werden. Wie es zu einem Anreiz von Kunden für die Nutzung solcher Geräte kommen könnte, soll später noch genauer erläutert werden.

Im Simulationsprogramm wird der DSM-Chip durch eine eigenständige Klasse mit dem Namen `DSM_Device` repräsentiert. Die Parameter, Attribute und Methoden dieser Klasse können in zwei Typen unterschieden werden, die im Folgenden als „DSM-Input“ und „DSM-Output“ bezeichnet werden. Diese werden in den nächsten Abschnitten genauer erläutert.

Bei der Erzeugung einer Instanz der Klasse `DSM_Device` wird ähnlich vorgegangen wie bei den Gerätklassen, repräsentiert durch die Superklasse `Device`. Die Parameter werden genauso in Dreier-Gruppen übergeben und gemäß der angegebenen Standardabweichung normalverteilt. Danach erfolgt die Auswahl der Funktionsweise des Chips, die durch die Parameter `DSM_Input_Mode` und `DSM_Output_Mode` erfolgt.

Bei der Implementierung war höchstmögliche Flexibilität der Definition des Einsatzes von verschiedenen Funktionsweisen des DSM-Chips ein vorrangiges Ziel, um deren Auswirkungen auf die Leistungsaufnahme von Geräten untersuchen zu können. Es ist daher möglich, Verteilungen der einzelnen Funktionsarten zu definieren, womit dann jede Instanz von `DSM_Device` individuell mit unterschiedlichen Kombinationen von Funktionen für „DSM-Input“ und „DSM-Output“ ausgestattet wird.

Jedes Gerät erhält bei der Instanzenerzeugung eine Instanz von `DSM_Device` zugewiesen (Attribut `DSM`, siehe z.B. Tabelle 2.3), welche, im einfachsten Fall keine DSM-Funktion aufweist. Um die Reaktion des Gerätes mit und ohne einen DSM-Chip in einem Simulationsdurchlauf ermitteln zu können, wird vor dem Start der Simulation der Instanz von `DSM_Device` eine „Kopie“ des Gerätes zur Verfügung gestellt, auf das es einwirken kann (Attribut `Controlled_Device` der Klasse `DSM_Device`, siehe Abbildung 2.4 auf Seite 51). Die originale Geräteinstanz bleibt von diesen Änderungen verschont. Es stehen somit beim Simulationsdurchlauf zwei identische Geräte zur Verfügung, von denen eines von einem DSM-Chip beeinflusst wird.

2.3.1. DSM-Input

Die Funktionen, die zum DSM-Input gehören, stellen die bereits erwähnten Funktionen des DSM-Chips dar. Sie ermitteln, ob eine Änderung des Geräteverhaltens notwendig ist und stellen diese Information über die Schnittstelle dem zu steuernden Gerät zur Verfügung. Zu diesem Zweck gibt es die beiden Attribute `Output` und `Warning`. Diese können als zwei Signalleitungen aufgefasst werden, auf denen über einen

Pegel der logische Zustand (vorhanden oder nicht) dieser Attribute zur Verfügung gestellt wird. Die Schnittstelle zwischen DSM-Chip und Gerät ist also sehr simpel.

In Tabelle 2.14 sind die wichtigsten Eigenschaften der Klasse `DSM_Device`, die zum DSM-Input gehören, aufgelistet. Über die Parameter wird die Funktion des DSM-Inputs bestimmt (siehe Abschnitt 2.1.3). Bei der Instanzenerzeugung wird aus den angegebenen DSM-Moden (`DSM_Input_Mode`) gemäß der angegebenen Verteilung ein bestimmter Modus ausgesucht, und die Instanz von `DSM_Device` auf diesen hin konfiguriert.

	Name	Einheit	Beschreibung
Parameter	Frequency_Level	Hz	Frequenzwert, ab dem es zu einer Änderung des Geräteverhaltens kommen soll
	Check_former_Frequ_Data	-	sollen zurückliegende Frequenzdaten überprüft werden? (wichtig für die korrekte Reaktion einiger Geräteklassen bei unterschiedlichen Zeitauflösungen, erhöht jedoch die Simulationszeit)
	Frequency_Hysteresis	Hz	Breite der Hysteresis um die Frequenz, z.B.: Output = 1 bei Frequenzwert Frequency_Level Output = 0 bei Frequenzwert Frequency_Level + Frequency_Hysteresis
	Frequ_Filter_Time	min	Gibt die Zeitspanne an, wie weit zurückliegende Frequenzwerte für die laufende Mittelwertbildung herangezogen werden sollen.
	Time_Delay_Restore_Op	min	Zeitspanne, um die eine Rückkehr zur normalen Betriebsweise verzögert werden soll
	Prioritys_Number	-	Anzahl an Prioritätsgruppen von Verbrauchern
	Prioritys_Frequ_Range	Hz	Frequenzbereiche der einzelnen Prioritätsgruppen
	Prioritys_Allocation	%	Verteilung der Verbraucher auf die Prioritätsgruppen
	DSM_Input_Mode	-	Liste mit den möglichen DSM-Input-Funktionen sowie deren Verteilung
	Last_Frequency_Data	-	Speicher der letzten Frequenzdaten (für laufenden Mittelwert)
Attribute	Time_Output_Start	-	Gibt den Zeitpunkt an, zu dem der Output des DSM-Chips für eine Verhaltensänderung aktiv wurde.
	Time_Output_Stop	-	Gibt den Zeitpunkt an, zu dem der Output des DSM-Chips zurückgesetzt wurde.
	input_algorithm	-	Funktionenhandle auf die zu verwendende Input-Funktion (Reaktion auf Frequenzlevel durch Aufruf der Methode <code>algorithm</code>)
DSM_Input_Mode	No_DSM_Function	-	keine DSM-Funktion
	Frequency_Response_Simple	-	Einfache Schaltschwelle anhand von Netzfrequenz
	Frequency_Response_Priority	-	Frequenzschaltwellen mit verschiedenen Prioritätsgruppen
Ausgabe	Output	-	Gibt an, ob durch das <code>DSM_Device</code> die Notwendigkeit ermittelt wurde, das Verbraucherverhalten zu beeinflussen
	Warning	-	Gibt an, ob durch das <code>DSM_Device</code> ein bedenklicher Netzzustand ermittelt wurde, auf den das Gerät eventuell reagieren sollte.

Tabelle 2.14: Zusammenfassung der Parameter, Attribute, Moden und Ausgaben des Typs „DSM-Input“ der Klasse `DSM_Device`

Tabelle 2.15 gibt ein Beispiel für die Definition der einzelnen DSM-Input-Moden für einen Gerätetyp. Die Prozentzahlen geben an, wie viele der Geräte dieses Gerätetyps welchen DSM-Input-Modus aufweisen. Im Beispiel haben 30% der Geräte keinen DSM-Chip eingebaut.

DSM - Settings:						
	DSM_Input_Mode					
	+--->	No_DSM_Function	-->	30	%	
	+--->	Frequency_Response_Simple	-->	50	%	
	'--->	Frequency_Response_Priority	-->	20	%	

Tabelle 2.15: Beispiel für die Definition der Verteilung der DSM-Input-Moden (Auszug aus Parameterdatei)

Bei der Simulation werden durch Aufruf der Funktion `algorithm` der `DSM_Device`-Instanz aktuelle Messdaten, wie z.B. die Netzfrequenz vom Simulationsprogramm übergeben und anhand dieser wird die Notwendigkeit eines Eingreifens ermittelt.

In den folgenden Abschnitten werden die verschiedenen DSM-Input-Moden genauer erläutert.

2.3.1.1. No_DSM_Function

Soll die `DSM_Device`-Instanz keinen Einfluss auf das Geräteverhalten haben, wird dieser DSM-Modus ausgewählt. Dies soll ein Gerät ohne einen DSM-Chip repräsentieren.

Mit Hilfe dieses DSM-Modus ist es möglich, in einer Gerätekonstellation einen gewissen Prozentsatz der Geräte ohne eine DSM-Funktionalität zu betreiben.

2.3.1.2. Frequency_Response_Simple

Das Verhalten des zu steuernden Gerätes wird in diesem Modus dann beeinflusst, wenn die Netzfrequenz um einen bestimmten Wert, definiert durch den Parameter `Frequency_Level`, eingebrochen ist. Durch die Streuung dieses Parameters hat jede `DSM_Device`-Instanz einen individuellen Frequenzwert, bei der diese reagiert. Diese Streuung kann zweierlei Hintergründe haben. Zum einen repräsentiert diese Streuung jene Ungenauigkeiten der Messungen, die sicherlich jeder Chip aufgrund von möglichst einfachen Messmethoden aufweisen wird. Zum anderen kann eine Streuung des Geräteverhaltens erzielt werden, indem absichtlich für jeden Chip bei der Erzeugung der Frequenzlevel anders definiert wird. Damit sollen bei einem Frequenzeinbruch nicht alle Geräte gleichzeitig reagieren. Auf diese Möglichkeit wird später bei den Simulationsergebnissen in Kapitel 5 noch einmal genauer eingegangen.

In Diagramm 2.16 zeigt sich die Auswirkung von verschiedenen Werten für die Standardabweichung anhand eines einfachen Beispiels. Hierbei hat die Frequenz einen linearen Verlauf. Von 1:00 bis 4:00 Uhr fällt sie linear um 2 Hz, bleibt konstant bis sie von 7:00 bis 10:00 Uhr wieder auf 50 Hz steigt.

<code>Check_former_Frequ_Data</code>	0			
<code>Frequency_Level</code>	1	Hz	20	%
<code>Frequency_Hysteresis</code>	0,35	Hz	10	%
<code>Frequ_Filter_Time</code>	15	min	10	%
<code>Time_Delay_Restore_Op</code>	25	min	20	%

Tabelle 2.16: Beispiel für Definition der Parameter für `Frequency_Response_Simple` (Auszug aus Parameterdatei)

Der Simulation liegen die Einstellungen, die in Tabelle 2.16 angegeben sind, zugrunde. Nur die Standardabweichung von `Frequency_Level` wurde variiert und es wurde kein Wert für `Frequency_Hysteresis`, `Frequ_Filter_Time` und `Time_Delay_Restore_Op` angegeben. Daher sollte bei einem Frequenzabfall von 1 Hz zumindest die Hälfte aller Geräte zu einer Leistungsreduktion beitragen. In Diagramm 2.16 ist dies gut zu erkennen. Weiters sieht man sehr gut, dass bei einer Erhöhung der Standardabweichung die Breite des Frequenzbereichs, in dem die DSM-Geräte reagieren, zunimmt.

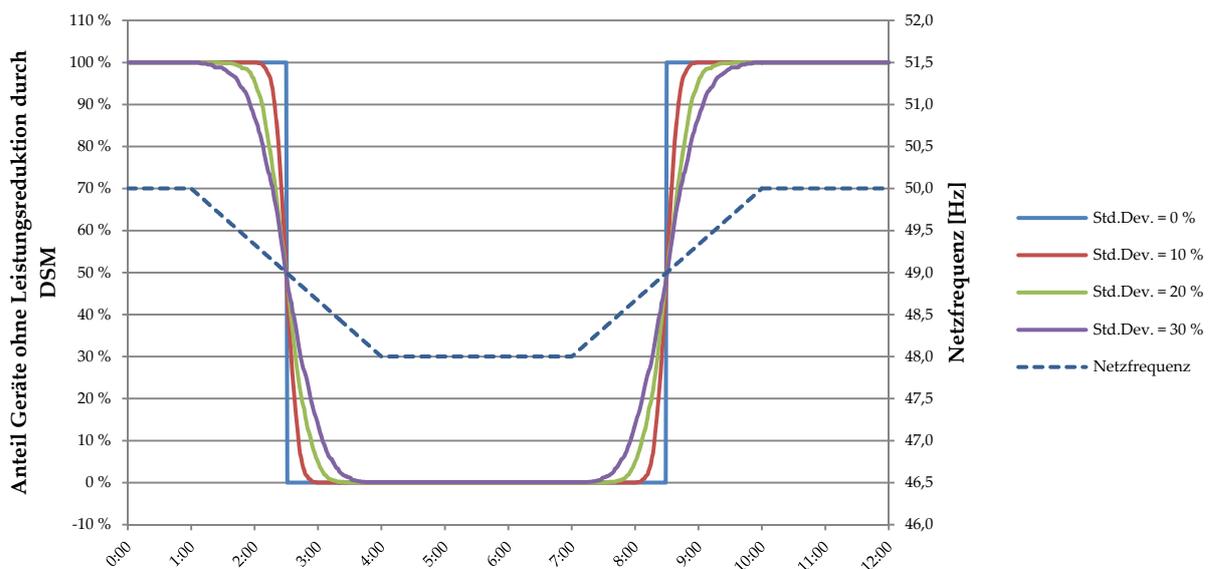


Diagramm 2.16: Auswirkung verschiedener Standardabweichungen des Frequenzschwellwertes auf die DSM-Funktion. Daten von 1.000 Verbrauchern

2.3.1.2.1. Behandlung von verrauschten Frequenzdaten

Wird kein Wert für `Frequency_Hysteresis` angegeben, ist die Reaktionsschwelle eine unmittelbare. Bei idealisierten Frequenzverläufen, die keinerlei Rauschen durch Mess-, Schalt- und Regelvorgänge aufweisen, stellt dies eine Vereinfachung der Berechnungen dar. In der Realität wird jedoch die Frequenz verrauscht sein, wodurch es bei einer scharfen Reaktionsschwelle auch zu einem stark „verrauschten“ Verhalten der Verbraucher kommt.

In Diagramm 2.17 und Diagramm 2.18 kann dies beobachtet werden. Hier wurde der Frequenzverlauf aus Diagramm 2.16 mit starkem Rauschen (Rauschamplitude von 0,25 Hz) beaufschlagt. Bei einer unmittelbaren Reaktionsschwelle reagieren die Verbraucher auch unmittelbar auf die Frequenzschwankungen (rote Kurve), wodurch sich eine stark schwankende Leistungsaufnahme ergibt. Dieses Verhalten ist natürlich nicht erstrebenswert, da es zu starken Belastungen einerseits des Netzes und andererseits der Verbraucher führt.

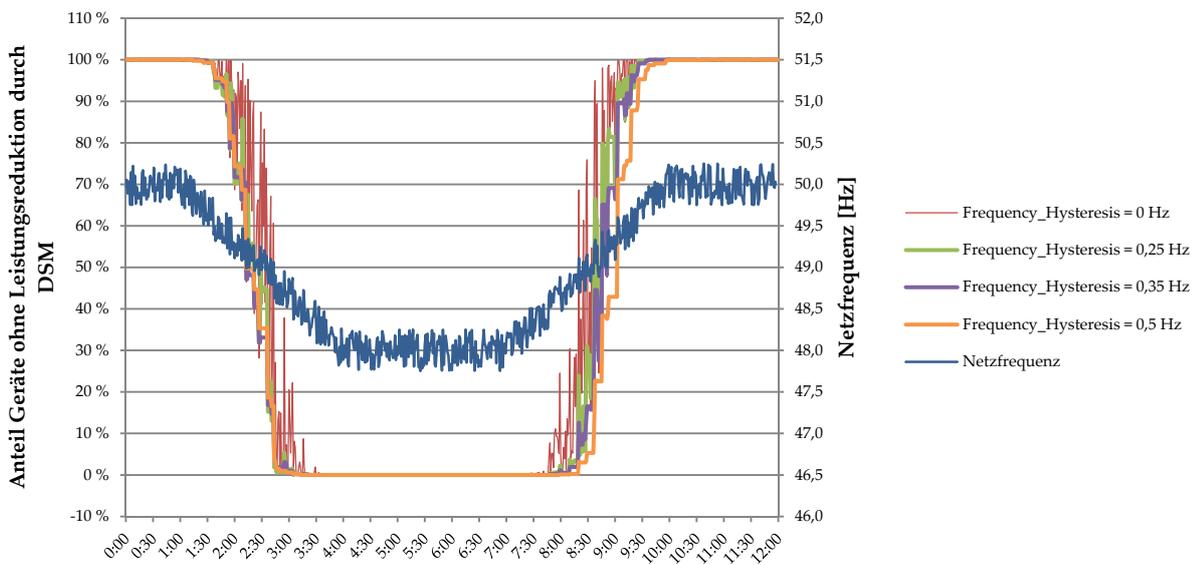


Diagramm 2.17: Auswirkungen verschiedener Breiten der Reaktionshysterese bei verrauschten Frequenzdaten, Daten von 1.000 Verbrauchern

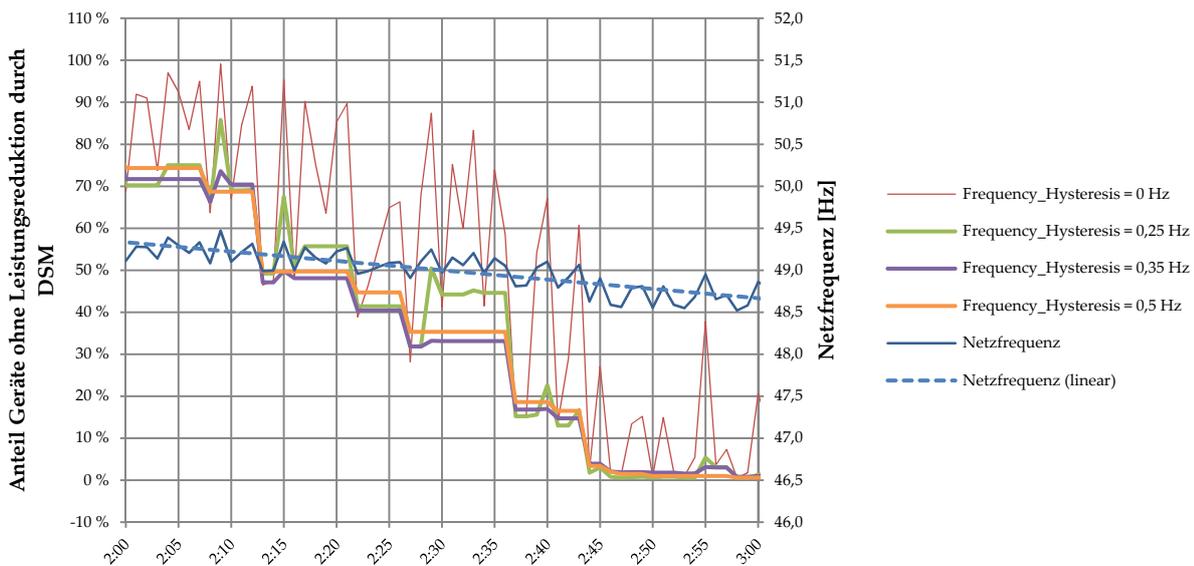


Diagramm 2.18: Detail aus Diagramm 2.17

Eine Möglichkeit, dieses Verhalten zu bekämpfen, liegt in der Definition einer Hysterese der Reaktionsschwelle. Dies erfolgt über den Parameter `Frequency_Hysteresis`. Die „Einschaltsschwelle“ (Beginn der Leistungsbeeinflussung) wird nach wie vor durch `Frequency_Level` bestimmt, die „Ausschaltsschwelle“ (Ende der Leistungsbeeinflussung) ist `Frequency_Level + Frequency_Hysteresis`. Je breiter die Hys-

terese desto stetiger wird das Verhalten der Verbraucher. Ist die Breite der Hysterese so groß wie die doppelte Rauschamplitude, was bedeutet, dass alle Abweichungen der Frequenz durch Rauschen in der Hysterese enthalten sind, wird der Verlauf des Verhaltens monoton fallend bzw. steigend (gelbe Kurve). Bei der Rückkehr der Netzfrequenz zu ihrem Ausgangswert bewirkt eine Hysterese jedoch, dass die Verbraucher erst später in ihren normalen Betriebszustand zurückkehren, wie z.B. deutlich in Diagramm 2.17 zu erkennen ist.

Eine weitere Möglichkeit liegt in der Filterung der Netzfrequenz mit einem Tiefpassfilter in Form einer laufenden Mittelwertbildung der Frequenzwerte. Dies könnte direkt der DSM-Chip vornehmen, um so nur auf langsame Änderungen der Netzfrequenz zu reagieren. Die Zeitspanne, in der die Frequenzwerte gemittelt werden sollen, wird durch `Frequ_Filter_Time` festgelegt.

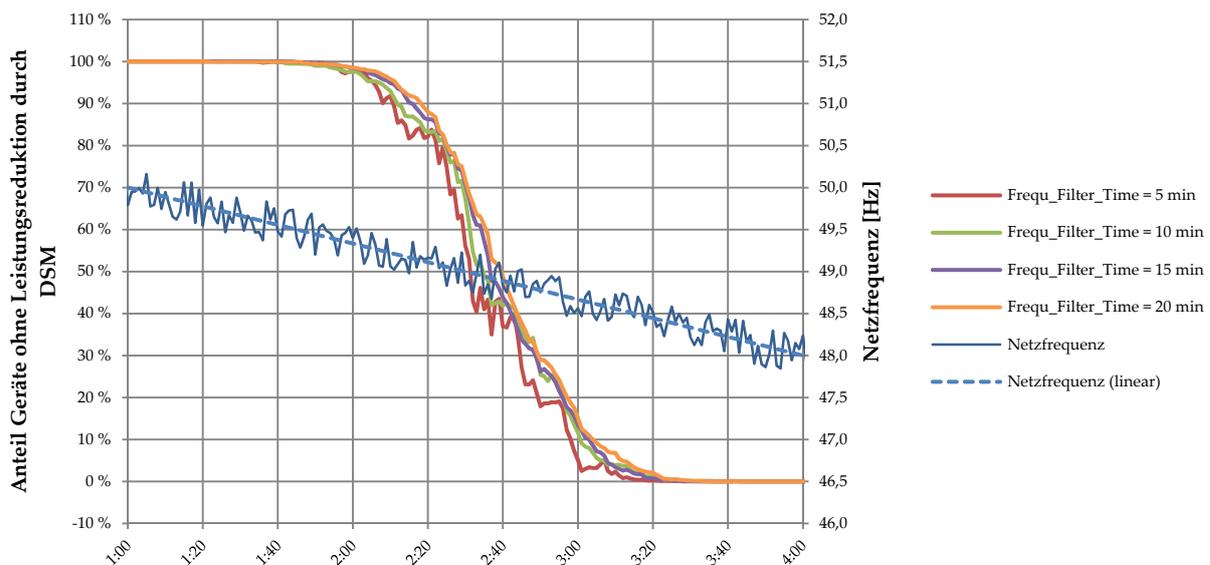


Diagramm 2.19: Auswirkungen verschiedener Zeiträume der laufenden Mittelwertbildung bei verrauschten Frequenzdaten, Daten von 1.000 Verbrauchern

Diagramm 2.19 zeigt die Auswirkung von verschiedenen Werten für `Frequ_Filter_Time`. Je höher der Wert dieses Parameters desto mehr Frequenzwerte werden zur Mittelwertbildung herangezogen und damit das Rauschen stärker unterdrückt sowie das Verhalten der Verbraucher immer stetiger. Durch die Filterung kommt es jedoch zu einer Zeitverzögerung der Reaktion der Verbraucher, was deutlich an der gelben Kurve in Diagramm 2.19 zu erkennen ist. Eigentlich sollten, da hier die gleichen Parameterwerte wie in Tabelle 2.16 angegeben wurden, die Hälfte aller Verbraucher bei 49 Hz zur Leistungsreduktion beitragen (um 2:30 Uhr). Durch die Filterung passiert dies um einige Minuten später.

Es ist auch eine Kombination aus Filter und Hysterese möglich, wodurch das Ergebnis noch mehr verbessert werden kann.

Um die Rechenzeit zu verringern ist es einfacher, idealisierte, rauschfreie Frequenzdaten anzunehmen, um so die Auswirkungen der DSM-Mechanismen zu untersuchen, wodurch die Notwendigkeit zur Filterung bzw. Einführung einer Hysterese wegfällt. In den meisten folgenden Simulationen wurden daher diese hier vorgestellten Mechanismen nicht angewendet.

2.3.1.2.2. Verzögern der Rückkehr zum Normalbetrieb

Mit dem Parameter `Time_Delay_Restore_Op` kann eine Verzögerungszeit eingestellt werden, die nach einer Erholung der Netzfrequenz vergehen soll, bis das Gerät wieder in einen normalen Betriebsmodus zurückkehrt. Im Prinzip wird durch die DSM-Einheit der ungünstige Netzzustand länger angezeigt. Dies soll helfen, Leistungsspitzen zu vermeiden, die sich bei der Reaktivierung vieler Geräte ergeben würde.

Dies ist nur in jenen Fällen anwendbar, in denen die Leistungsbeeinflussung des Gerätes durch den Benutzer kaum wahrgenommen werden kann bzw. die Gerätefunktion nicht zu weit beeinträchtigt wird. Ein Beispiel dafür wäre z.B. ein Kühl- oder Gefrierschrank, dessen Soll-Temperatur auf einen Wert angehoben

wurde, der auch bei längerem Vorherrschen zu keiner Beeinträchtigung der Haltbarkeit der Lebensmittel führt.

In Diagramm 2.20 sind die Auswirkungen bei Einführung dieses Parameters dargestellt. Dabei wurden die gleichen Geräte herangezogen, die auch für die Darstellung in Diagramm 2.16 verwendet wurden. Die Standardabweichung der Frequenzschwelle liegt immer bei 0%.

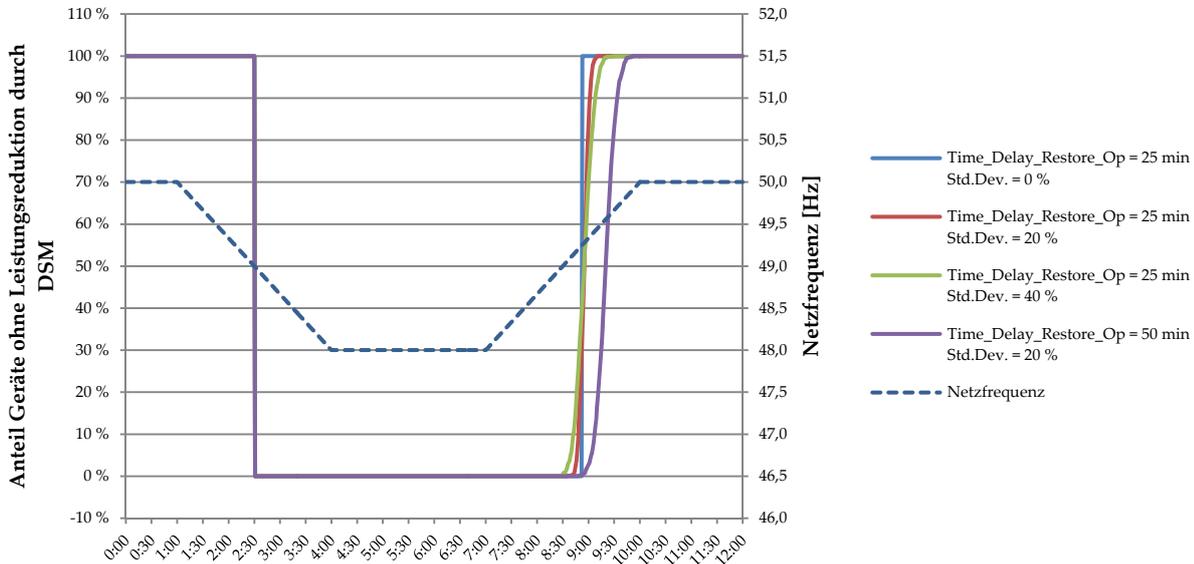


Diagramm 2.20: Auswirkungen des Parameters `Time_Delay_Restore_Op` auf die Leistungsreduktion.

Bei einer Verzögerungszeit von 25 min, die ohne Standardabweichung angegeben wird (blaue Kurve), kommt es nicht um 8:30 Uhr sondern erst um 8:55 Uhr zu einer Rückkehr zur normalen Geräteoperation. Bei einer Angabe höherer Standardabweichungen wird der Verlauf dementsprechend verschliffen (rote und grüne Kurve). Durch eine Erhöhung der Verzögerungszeit wird der Zeitpunkt des Rückkehres zur normalen Betriebsweise weiter hinausgezögert (violette Kurve).

Zwar kann über die Streuung der Frequenzschwelle auch die Gleichzeitigkeit des Geräteverhaltens reduziert werden (siehe Diagramm 2.16), allerdings nur in einem beschränkten Umfang, welcher durch den realen Verlauf der Netzfrequenz vorgegeben ist. Mit Hilfe des Parameters `Time_Delay_Restore_Op` kann hier eine breitere Streuung realisiert werden.

2.3.1.2.3. Die *look_back*-Funktionen

Für viele Simulationssituationen ist es nicht notwendig, zu wissen, wann genau die Frequenzschwelle unterschritten wurde. Es ist oft ausreichend zu wissen, ob dies für den aktuellen Beobachtungszeitpunkt gilt (z.B. für eine Leistungsreduktion eines Leuchtkörpers oder das Ausschalten eines Gerätes).

In manchen Situationen muss jedoch dieser Zeitpunkt genau ermittelt werden, da die Zeitspanne, die von da an verstreicht, bereits das Verhalten des Gerätes zum Zeitpunkt der Beobachtung beeinflusst hat. Insbesondere fällt dies ins Gewicht, wenn unterschiedliche Simulationsauflösungen (und damit Beobachtungszeitpunkte) gewählt werden.

Um dies zu berücksichtigen, können mit Hilfe der Option `Check_former_Frequ_Data` (1 = aktiv) jene Gerätetypen gekennzeichnet werden, die in dieser Hinsicht eine besondere Behandlung erfordern.

Ist `Check_former_Frequ_Data` aktiv, werden die sogenannten *look_back*-Funktionen für alle DSM-Funktionalitäten herangezogen. Diese ermitteln bzw. erhalten die Information, wann genau die Beeinflussung des Geräteverhaltens gestartet und gestoppt wird und sorgen dafür, dass sich das Geräteverhalten, unabhängig von der aktuellen Simulationsauflösung, punktgenau im richtigen Moment ändert. Diese Zeitinformationen werden durch die jeweiligen Input-Funktionen dadurch ermittelt, dass bei der Funktion `algorithm` zum Zeitpunkt, zu dem sie aufgerufen wird, nicht nur der aktuelle Frequenzwert dieses Zeitpunktes untersucht wird, sondern auch alle Frequenzwerte seit dem letzten Simulationsschritt (Zurückschauen, „look back“). Diese Daten werden vom Simulationsprogramm der Funktion `algorithm` zur Ver-

fügung gestellt. Mit Hilfe dieser Frequenzwerte wird der exakte Zeitpunkt der Beeinflussung ermittelt, und in den Attributen `Time_Output_Start` (Zeitpunkt des Beginns der Verhaltensänderung) und `Time_Output_Stop` (Zeitpunkt, zu dem die Leistungsreduktion nicht mehr durchgeführt werden muss) gespeichert. Dadurch wird erreicht, dass das Geräteverhalten unabhängig von den gewählten Simulationszeitschritten ist.

Diese Option ist also keine Eigenschaft des DSM-Chips, sondern ist nur notwendig, um eine fehlerfreie Simulation zu gewährleisten. Jene Gerätetypen, bei denen dieser höhere Aufwand nicht notwendig ist, werden mit einfacheren Funktionen simuliert, womit der Rechenaufwand reduziert werden kann.

Vorgreifend soll ein Beispiel für die Notwendigkeit der Unterscheidung mit der Option `Check_former_Frequ_Data` gegeben werden. Dazu eignen sich die Gerätetypen `Thermal_Storage` mit einer Temperaturänderung einerseits und `Probable_Operation` mit einer Leistungsreduktion andererseits.

Bei der Klasse `Probable_Operation` ist die Vergangenheit der Frequenzdaten nicht wichtig, da die aktuelle Leistungsaufnahme nur zum Beobachtungszeitpunkt ermittelt werden muss und durch den DSM-Chip entschieden wird, ob die Leistung reduziert werden soll oder nicht (siehe Abschnitt 2.3.2.3).

Wird hingegen bei der Klasse `Thermal_Storage` die Solltemperatur verändert, ist zum Beobachtungszeitpunkt wichtig, ob dies nun vor fünf oder 60 Minuten passiert ist, da sich in der Zwischenzeit die Temperatur und die Zufuhr thermischer Energie anders verhalten haben als mit den ursprünglichen Einstellungen. Bei einer feinen Simulationsauflösung fällt dies nicht ins Gewicht. Bei verringerter Auflösung wird der Fehler aber immer gravierender. Eine genaue Gegenüberstellung dieses Effektes gibt Abschnitt 2.3.2.4 und Abschnitt 2.3.2.5.

2.3.1.3. Frequency_Response_Priority

Bei einem simplen Frequenzverhalten des DSM-Chips hat der Benutzer keine Möglichkeit, dieses Verhalten zu beeinflussen, da davon ausgegangen werden kann, dass die Eigenschaften des Chips nach der Herstellung nicht mehr geändert werden können.

Um aber dem Benutzer die Wahl zu lassen, welche Geräte zuerst in ihrem Verhalten beeinflusst werden sollen, falls die Netzsituation dies erforderlich macht, könnten verschiedene Prioritätsgruppen definiert werden. Der Benutzer wählt für jedes Gerät die Priorität, mit der es behandelt werden soll. Ob dies nun über einen Austausch des gesamten DSM-Chips (ähnlich einer Decoder-Karte) oder durch eine Einstellmöglichkeit am Gerät selbst erfolgt (z.B. durch einen DIP-Schalter), ist Sache der hardwaremäßigen Umsetzung.

Es könnte dann so sein, dass z.B. der Kunde eines Netzbetreibers mit Preisnachlässen belohnt wird, wenn er Geräte betreibt, die bei Netzproblemen früh ihren Energieverbrauch zurücknehmen. Für jedes Gerät, das sich weniger kooperativ verhält, könnten zusätzliche Zahlungen notwendig sein. So könnte erreicht werden, dass die Verbraucher mit den verschiedenen Prioritätsgruppen betrieben werden.

Mit dem Parameter `Prioritys_Number` wird definiert, wie viele unterschiedliche Prioritätsgruppen es geben soll, `Prioritys_Frequ_Range` gibt an, wie breit der Frequenzbereich ist, in dem die einzelnen Prioritätsgruppen im Betriebsverhalten beeinflusst werden. `Prioritys_Allocation` beinhaltet die Verteilung der Geräte eines Gerätetyps auf die einzelnen Prioritätsgruppen. Alle diese Parameterwerte werden jeweils mit abfallender Priorität von links nach rechts aufgelistet, das heißt ganz links ist die höchste Priorität zu finden.

Ein Beispiel für eine Definition von Prioritätsgruppen gibt Tabelle 2.17. Es werden darin drei Prioritätsgruppen definiert. Bei einer Frequenzabweichung von 0,5 Hz würden 20 % der Geräte ihre Leistung reduzieren (niedrigste Priorität), ab einer Abweichung von 0,8 Hz weitere 50 % wohingegen die restlichen Geräte (DSM-Funktion vorausgesetzt) erst bei einer Abweichung von 1 Hz reagieren würden (höchste Priorität).

<code>Prioritys_Number</code>	3			
<code>Prioritys_Frequ_Range</code>	0,2	0,3	0,5	Hz
'-> Std.Dev.:	15	%		
<code>Prioritys_Allocation</code>	30	50	20	%

Tabelle 2.17: Definition von Prioritätsgruppen (Auszug aus Parameterdatei)

Hier sorgt auch die Angabe einer Standardabweichung dafür, dass jede `DSM_Device`-Instanz individuelle Reaktionen zeigt.

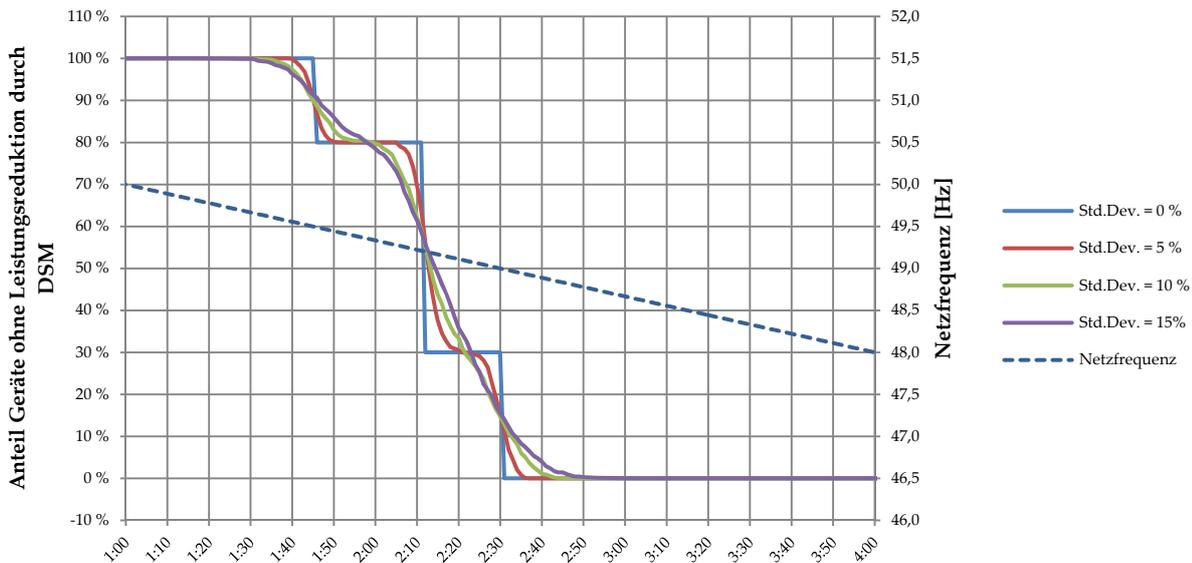


Diagramm 2.21: Auswirkung verschiedener Standardabweichungen der Frequenzschwellwerte auf die DSM-Funktion beim Modus `Frequency_Response_Priority`. Parameter nach Tabelle 2.17, Daten von 1.000 Verbrauchern

Diagramm 2.21 zeigt das Verhalten von Verbrauchern, die gemäß Tabelle 2.17 in Prioritätsgruppen eingeteilt wurden. Sehr gut ist die Reaktion bei einer Streuung der Frequenzwerte von 0% zu erkennen. Wie bereits beschrieben reduzieren zuerst die Verbraucher mit der niedrigsten Priorität ihre Leistung (20% bei 49,5 Hz um 1:45 Uhr), dann jene der 2. Gruppe (50 % bei 49,2 Hz um 2:12 Uhr) und zum Schluss die Geräte mit der höchsten Priorität (30% bei 49 Hz um 2:30 Uhr). Durch das Angeben einer Standardabweichung wird der Treppenverlauf dementsprechend verschliffen.

Der Input-Modus `Frequency_Response_Priority` ist in seiner Ausführung eng an den Modus `Frequency_Response_Simple` gebunden, da bei der `DSM_Device`-Instanzenerzeugung für jeden Chip aus den angegebenen Parametern ein einzelner Frequenzschwellwert ermittelt wird. Betrachtet man das Kollektiv, verhält sich dieses, wie in Diagramm 2.21 dargestellt.

Das Einzelgerät besitzt jedoch nur einen einzigen Schwellwert und verwendet dementsprechend alle Funktionen, die auch im Modus `Frequency_Response_Simple` verwendet werden. Deshalb gelten alle Aussagen aus Abschnitt 2.3.1.2 auch hier, insbesondere jene zur Verwendung der Parameter `Frequency_Hysteresis`, `Frequ_Filter_Time`, `Time_Delay_Restore_Op` und `Check_former_Frequ_Data`, die genauso in diesem Input-Modus definiert werden können.

2.3.2. DSM-Output

Die Funktionsweise des DSM-Chips ist eigentlich durch die Funktionen des Typs `DSM-Input` vollständig implementiert. Als Ausgabe liefern diese Funktionen die Attribute `Output` und `Warning`, die die Schnittstelle des Chips mit dem Gerät darstellen sowie, falls notwendig, die Werte für `Time_Output_Start` und `Time_Output_Stop` für die Sicherstellung von korrekten Simulationsergebnissen.

Die Funktionen von `DSM-Output` stellen nun die Reaktion des Gerätes auf die Signale des `DSM-Chips` dar. Eigentlich sollten diese Funktionen in den jeweiligen Geräteklassen implementiert sein, da sie ja Funktionalitäten eines bestimmten Gerätes darstellen. In der in dieser Arbeit gemachten Implementierung des Simulationsprogramms wurden jedoch aufgrund der einfacheren Umsetzung und des verringerten Rechenaufwandes diese Funktionen in der Klasse `DSM_Device` angesiedelt.

Dadurch ist es zum einen möglich, aus einem Pool von möglichen Funktionen auszuwählen, die das Geräteverhalten beeinflussen. Bei der Instanzenerzeugung jedes `DSM_Device` wird dann eine Funktion konkret herangezogen und über Funktionen-Handles während der Simulation angesprochen. So muss die Auswahl der Output-Funktion und deren Parameter nicht während der Simulation für jeden Simulationsschritt ausgeführt werden.

Weiters werden so alle Parameter, die mit DSM zu tun haben, von einer Klasse verarbeitet und gespeichert, was deren Handhabung deutlich vereinfacht, als wenn sie über mehrere Geräteklassen aufgeteilt würden.

Ähnlich wie bei den Parametern der Moden für den DSM-Input werden auch jene für den DSM-Output in der Parameterdatei angegeben (siehe Abschnitt 2.1.3). Auch werden die Parameter genauso bei der Instanzerzeugung gemäß der angegebenen Standardverteilung gestreut.

Die Funktionen von DSM-Output verwenden, um nochmal das Charakteristikum der Schnittstelle hervorzuheben, nur die Attribute `Output` und `Warning`, sowie alle Daten, die den Geräteinstanzen zur Verfügung stehen. Anhand dieser Werte wird dann je nach Anforderung in das Geräteverhalten der Kopie der Geräteinstanz `Controlled_Device` eingegriffen. Die Notwendigkeit dieser Kopie ergibt sich aus zwei Gründen: Einerseits kann so das unbeeinflusste Verhalten im selben Simulationsdurchlauf ermittelt werden. Andererseits ist es in manchen Fällen notwendig, gewisse Parameter der unbeeinflussten Geräteinstanz zur Verfügung zu haben, um z.B. Einstellungen wieder zurückzusetzen. Durch die Kopie müssen diese nicht einzeln gesichert werden.

	Name	Einheit	Beschreibung
Parameter	Temp_Set_Variation	°C	Gibt an, um welche Temperatur die Solltemperatur einer Instanz der Klasse <code>Thermal_Storage</code> geändert werden soll
	Power_Reduction	%	Gibt an, um welchen Prozentsatz die aufgenommene Leistung eines Gerätes reduziert werden soll bzw. kann
	Time_Postpone_Start	min	Um wie viel wird die Startzeit einer Instanz der Klasse <code>Loadcurve_Operation</code> verschoben, wenn ein DSM-Output vorliegt?
	Time_Postpone_max	min	Wie lange darf der Startzeitpunkt maximal verschoben werden?
	Consider_non_stop_Parts	-	Sollen die nicht unterbrechbaren Teile einer Lastkurve berücksichtigt werden
	DSM_Output_Mode	-	Liste mit den möglichen DSM-Input-Funktionen sowie deren Verteilung
Attribute	Controlled_Device	-	Kopie der Instanz des kontrollierten Gerätes. Hier erfolgen die Änderungen durch das DSM-System.
	Time_Postpone	-	Um wie viel Zeit wurde die Startzeit einer Instanz der Klasse <code>Loadcurve_Operation</code> bereits verschoben?
	output_next_step	-	Funktionenhandle auf Output-Funktion (Geräteverhalten bei DSM-Output)
	find_start_idx	-	Funktionenhandle auf Suchfunktion der Startzeiten für Lastkurven von Instanzen der Klasse <code>Loadcurve_Operation</code> , die innerhalb des betrachteten Zeitraums liegen
DSM_Output_Mode	postpone_schedule	-	Funktionenhandle auf die Verschiebefunktion eines Einsatzplanes von Instanzen der Klasse <code>Loadcurve_Operation</code> mit neuen Startzeiten
	Turn_Off	-	Ausschalten des Gerätes
	Turn_Off_Stand_by	-	Ausschalten des Gerätes, falls sich dieses im Stand-by-Betrieb befindet, ansonsten normale Operation
	Reduce_Input_Power	-	Leistungsreduktion der aufgenommenen Leistung
	Change_Temp_Set	-	Änderung der Solltemperatur einer Instanz der Klasse <code>Thermal_Storage</code>
	Postpone_Start	-	Verzögerung des Einschaltens einer Instanz der Klasse <code>Loadcurve_Operation</code>
	Pause_Programm	-	Lastkurve einer Instanz der Klasse <code>Loadcurve_Operation</code> wird in ihrer Ausführung unterbrochen
Ausgabe	Power_Input	W	Aufgenommene Leistung des Gerätes mit DSM

Tabelle 2.18: Zusammenfassung der Parameter, Attribute, Moden und Ausgaben des Typs „DSM-Output“ der Klasse `DSM_Device`

Die Ermittlung der Reaktion des Gerätes erfolgt, wie bei den Geräteinstanzen, in einer Funktion `next_step`, die meist einen Aufruf der gleichnamigen Funktion der Geräteinstanz `Controlled_Device` enthält. Bei manchen Output-Funktionen kann dieser Rechenschritt sogar entfallen, da bei jedem Simulationsschritt vor der Ermittlung der Daten mit DSM jene des unbeeinflussten Gerätes bereits vorliegen. Aus diesen lässt sich z.B. im Fall `Turn_Off` direkt die Leistungsreduktion ermitteln, ohne noch einmal einen Simulationsschritt mit der Geräteinstanz `Controlled_Device` durchführen zu müssen. Dies führt zu einer deutlichen Einsparung beim Rechenaufwand.

2.3.2.1. Turn_Off

Die einfachste Art der Leistungsreduktion bei einem Gerät besteht darin, einfach das Gerät vom Netz zu trennen, falls dies erforderlich ist.

Dies könnte entweder ein eingebundener DSM-Chip direkt im Gerät bewerkstelligen (das Gerät reagiert auf das Signal der Schnittstelle und betätigt selbst ein eigenes Hauptrelais) oder der DSM-Chip befindet sich in einem Zwischenstecker, in dem auch ein Relais verbaut ist, dass durch den Chip gesteuert wird. So ein Zwischenstecker könnte theoretisch vor jedes Gerät geschaltet werden. Eine weitere Möglichkeit wäre eine Einheit für die Montage im Sicherungskasten einer Elektro-Installation. So könnten z.B. die Beleuchtung und Steckdosen in verschiedene Gruppen unterteilt werden. Eine Gruppe ist ohne DSM-Funktion (z.B. je ein Beleuchtungskörper in jedem Raum, die als „Notbeleuchtung“ dienen), die andere Gruppe wird vom Netz genommen, falls ein kritischer Netzzustand erreicht wird (zusätzliche Beleuchtung, Steckdosen). Dies setzt natürlich eine dahingehende ausgeführte Installation voraus.

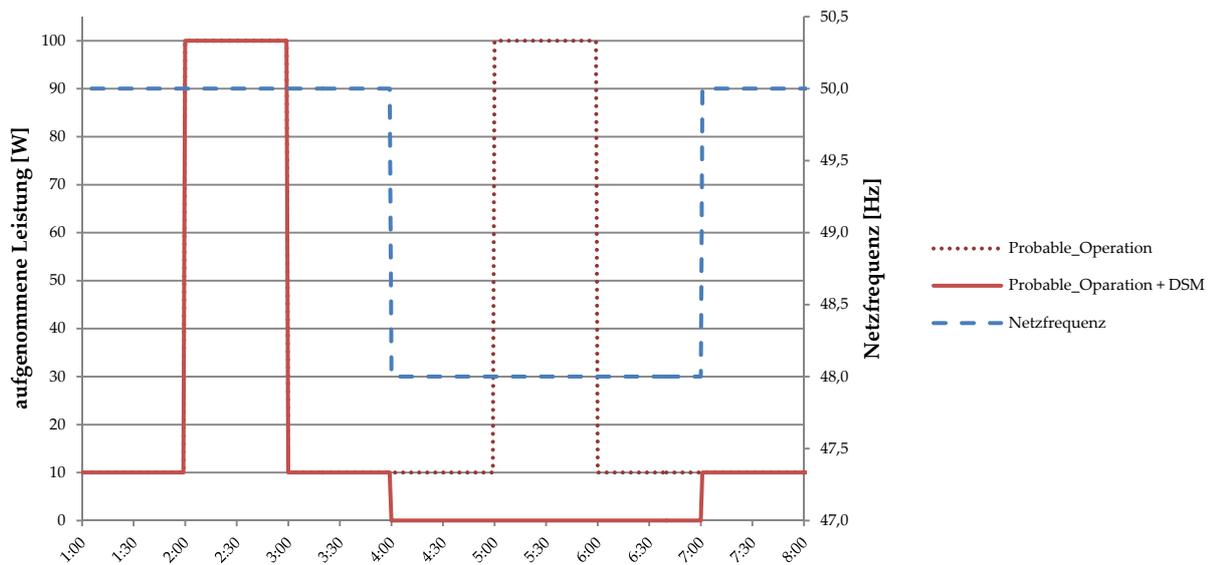


Diagramm 2.22: Turn_Off bei einem Gerät der Klasse Probable_Operation mit Stand-by-Verbrauch.

In Diagramm 2.22 ist die Auswirkung eines Gerätes mit der DSM-Reaktion Turn_Off zu sehen. Der Frequenzverlauf ist ein abrupter Einbruch um 4:00 und ein Anstieg um 7:00, weshalb es zwischen diesen Zeitpunkten zu einer Leistungsreduktion kommen muss. Das Gerät ist eine Instanz der Klasse Probable_Operation mit einem Stand-by-Verbrauch von 10 W und einer Betriebsleistung von 100 W. Deutlich ist zu erkennen, dass in der Zeit des Frequenzeinbruchs die aufgenommene Leistung den Wert 0 W annimmt. Sobald der kritische Netzzustand verlassen wird, kehrt das Gerät zu seiner normalen Betriebsweise zurück.

Dieser DSM-Output-Modus lässt sich auf alle Geräteklassen anwenden, auch wenn bei manchen Geräteklassen eine besondere Behandlung notwendig ist.

So wird für die Klasse Thermal_Storage nur die Höhe der Änderung der Solltemperatur (Parameter Temp_Set_Variation) auf $-\infty$ (bei Heizgeräten) bzw. $+\infty$ (bei Kühlgeräten) gesetzt und dann die Output-Funktion Change_Temp_Set verwendet (siehe Abschnitt 2.3.2.4). Dadurch wird das Abkühlen bzw. Erwärmen des thermischen Speichers während der Phase der Deaktivierung akkurat berücksichtigt, wenn das Gerät wieder in Betrieb genommen wird. In Diagramm 2.23 ist dies anhand eines Kühlgerätes dargestellt. Zusätzlich zur Leistung ist auch die Temperatur im Inneren des Speichers dargestellt. Deutlich ist zu erkennen, dass in der Zeit, in der die Kühlung nicht in Betrieb ist, sich die Temperatur immer mehr der Raumtemperatur (25°C) annähert. Wird das Gerät wieder aktiviert (um 12:00 Uhr), muss es deutlich länger laufen, um wieder die angestrebte Solltemperatur zu erreichen.

Bei einem Gerät der Klasse Loadcurve_Operation wird das laufende Programm einfach unterbrochen. Dies erfolgt mit Hilfe der Output-Funktion Pause_Programm (siehe Abschnitt 2.3.2.6), wobei die Parameter dementsprechend angepasst werden, damit sich das Gerät so verhält, als ob es vom Netz getrennt würde.

In Diagramm 2.24 sind zwei Beispiele für ein Turn_Off bei Instanzen der Klasse Loadcurve_Operation, die dem Beispiel aus Tabelle 2.13 in Abschnitt 2.2.7 entsprechen. Das erste Gerät startet um 3:00 (grüne Kurve) und das zweite um 3:45 Uhr (rote Kurve).

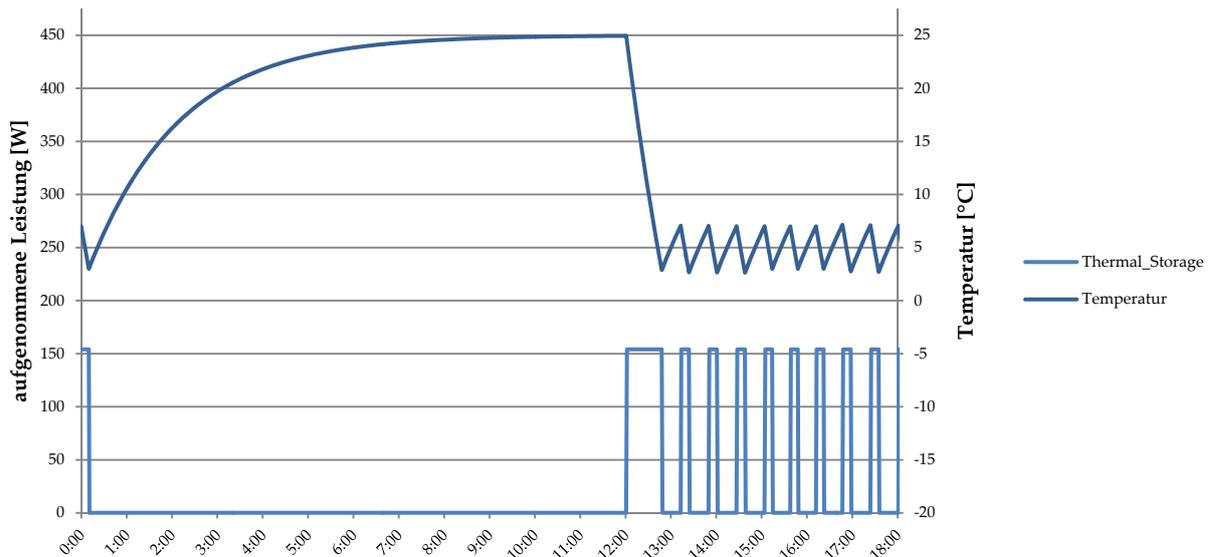


Diagramm 2.23: Turn_Off bei einem Gerät der Klasse `Thermal_Storage` (Kühlgerät). In der Zeit von 0:30 bis 12:00 bricht die Frequenz um 2 Hz ein (ähnlich Diagramm 2.22, nicht eingezeichnet)

Man kann erkennen, dass um 4:00 Uhr beide Geräte vom Netz getrennt werden und keine Leistung mehr aufnehmen. Der einzige Unterschied zwischen den beiden Geräten tritt beim Wiedereinschalten auf. Es wird nämlich von dieser Gerätekategorie angenommen, dass sie, falls das Programm bei einem Stromausfall unterbrochen wurde, dieses bei erneuter Energieversorgung fortführen und beenden. Bei modernen (nicht-flüchtiger Speicher der Elektronik) und sehr alten Geräten (elektromechanische Ablaufsteuerung) trifft diese Annahme durchaus zu. Alle anderen Geräte werden in ihrem Verhalten bei Trennung der Energieversorgung nicht berücksichtigt.

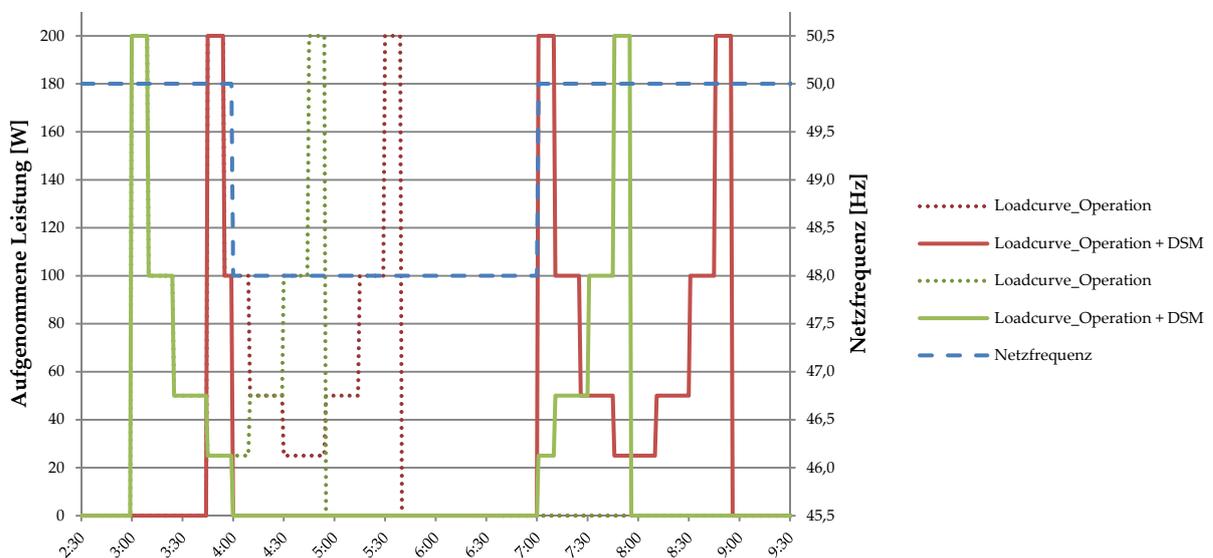


Diagramm 2.24: Turn_Off bei einem Gerät der Klasse `Loadcurve_Operation`

Bei der Definition der Lastkurve konnten Abschnitte definiert werden, die eigentlich nicht unterbrochen werden sollten (`Loadcurve_non_stop_parts`, siehe Abschnitt 2.2.6.1). Im Beispiel waren dies jeweils die drei höchsten Leistungsbereiche. Wird nun das Programm aufgrund eines Stromausfalls in so einem Bereich unterbrochen, wird dieser komplette Abschnitt der Lastkurve vom Gerät bei der Wiederkehr der Energieversorgung wiederholt. Dies soll simulieren, dass z.B. bei einer Waschmaschine die Waschlaube abgekühlt ist und wieder aufgeheizt werden muss. Ähnliches gilt auch bei einem Geschirrspüler. Dieser Fall ist bei der roten Kurve in Diagramm 2.24 dargestellt. Man kann erkennen, dass in diesem speziellen Fall sogar das gesamte Programm von vorne gestartet wird.

Bei der grünen Kurve kam es zu der Unterbrechung in einem Teil der Lastkurve, dessen Unterbrechung keine Folgen hat. Dies würde z.B. bei einer Waschmaschine der Spülvorgang sein, der einfach nach der Unterbrechung fortgesetzt werden kann.

Die Unterbrechung der Stromversorgung sollte bei den Geräteklassen `Thermal_Storage` und `Loadcurve_Operation` trotzdem eher nicht zur Anwendung kommen, da doch starke Einschränkungen der Funktion der Geräte gemacht werden. So ist es nicht ideal, wenn ein Kühlschrank, wie in Diagramm 2.23 dargestellt, stundenlang ausgeschaltet wird und sich die Lebensmittel unzulässig erwärmen. Auch ist es im Fall einer Waschmaschine für die Wäsche nicht vorteilhaft, wenn sie stundenlang in der Waschlauge liegt. Dennoch kann dieser DSM-Modus für diese Geräteklassen angewendet werden, um z.B. in Notfallsituationen ihren Beitrag zur Reduzierung der Leistung zu liefern. Für das häufiger genutzte DSM-Potential dieser Geräteklassen stehen andere Output-Moden zur Verfügung (`Change_Temp_Set`, `Postpone_Start` usw.).

2.3.2.2. Turn_Off_Stand_by

In diesem Modus wird im Fall eines kritischen Netzzustandes nur die Stand-by-Leistung eines Gerätes ausgeschaltet. Der normale Betrieb des Gerätes bleibt davon nicht betroffen. In Diagramm 2.25 ist dies dargestellt.

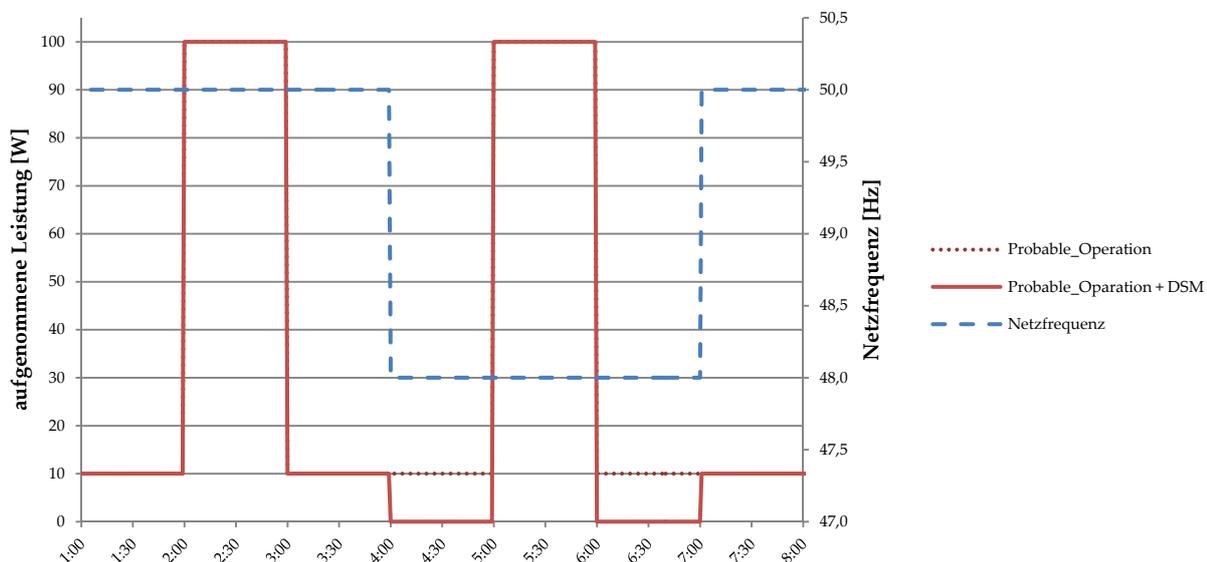


Diagramm 2.25: Turn_Off_Stand_by bei einem Gerät der Klasse Probable_Operation

Inwieweit dieser Modus eine realistische Möglichkeit zur Leistungsreduzierung darstellt, hängt sehr vom Gerät ab, das diesen Modus zur Verfügung stellt. Immerhin muss dieses den Stand-by-Verbrauch selbst abschalten und dennoch erkennen, wenn der Benutzer es verwenden will. Hat das Gerät diese Funktion bereits, spräche nichts dagegen, dass generell der Stand-by-Verbrauch auf dem niedrigen Niveau belassen wird, unabhängig vom Netzzustand, um Energie zu sparen.

Wenn aber der Stand-by-Verbrauch dazu dient, das Gerät schneller benutzbar zu machen, wie z.B. bei einem Laserdrucker, wäre eine Reduzierung dieser Leistung durchaus sinnvoll. Die einzige Einschränkung auf Benutzerseite ist eine etwas erhöhte Wartezeit.

Generell ist aber davon auszugehen, dass der Stand-by-Verbrauch von neuen Geräten immer mehr reduziert werden wird, da auch von Seiten der Europäischen Union dahingehende Richtlinien erlassen wurden [15] und daher dieses Potential zur Leistungsbeeinflussung immer geringere Bedeutung haben wird.

2.3.2.3. Reduce_Input_Power

Bei diesem Modus wird die aufgenommene Leistung des Gerätes um den Faktor `Power_Reduction` verringert. Diese Leistungsreduktion muss in den meisten Fällen das Gerät selbst erzielen, da der Einsatz von z.B. externen Dimmern nur bei einem sehr kleinen Prozentsatz von Geräten möglich ist. Hier sollte wieder

über die Schnittstelle die Information dem Gerät übermittelt werden, ob eine Leistungsreduktion notwendig ist, worauf dieses dann dementsprechend reagiert.

Geht man von einer hohen Stückzahl und damit niedrigen Kosten eines DSM-Chips aus, könnte dieser in fast jedem Gerät für eine Leistungsreduktion sorgen, angefangen bei Bildschirmen, Fernsehern bis hin zu einzelnen Beleuchtungskörpern. Natürlich muss eine dementsprechende Integration von den Geräteherstellern durchgeführt werden.

In Diagramm 2.26 ist die Leistungsreduktion an einem Gerät der Klasse `Probable_Operation` dargestellt. Bei verringerter Frequenz wird die Leistung um 10% reduziert. Auch ist zu erkennen, dass diese Reduktion nur auf die Betriebsleistung wirkt, der Stand-by-Verbrauch bleibt der gleiche. Dies berücksichtigt den Umstand, dass im Stand-by-Betrieb meist keine Leistungsreduktion möglich ist, da der Stand-by-Verbrauch entweder Leerlaufverluste von z.B. Netzteilen oder minimale, für das Erhalten eines Datenspeichers oder der Versorgung einer Uhr benötigte Leistung darstellen.

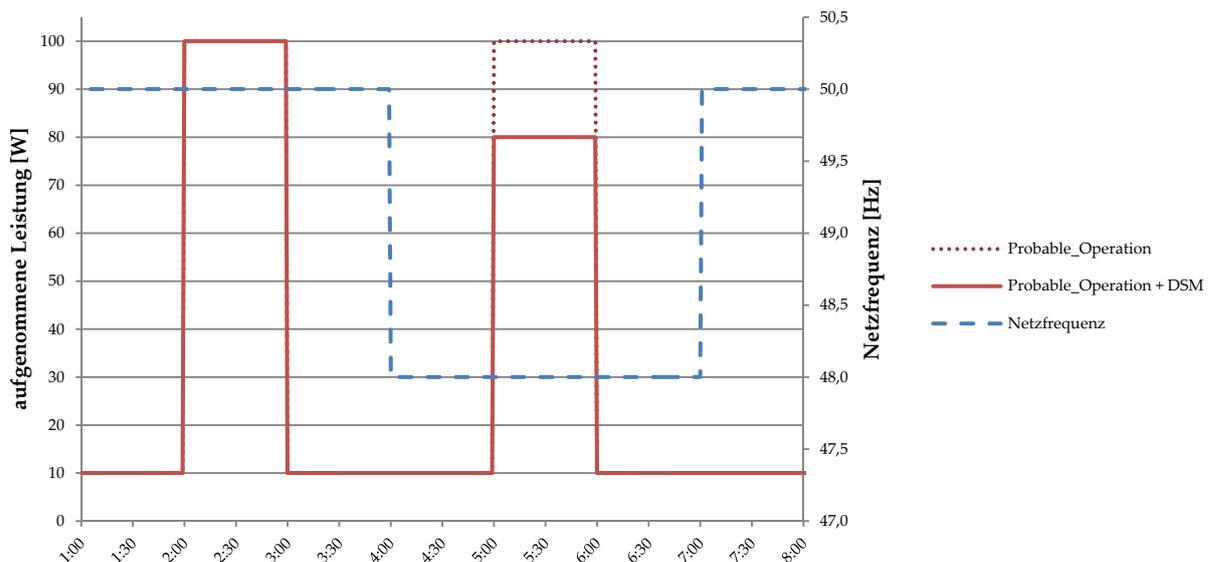


Diagramm 2.26: `Reduce_Input_Power` bei einer Instanz der Klasse `Probable_Operation`

Dieser Modus kann theoretisch auf alle Geräteklassen angewendet werden, für manche ist jedoch die Nutzung nicht zielführend oder möglich.

Bei der Klasse `Thermal_Storage` würde eine Leistungsreduktion zu längeren Laufzeiten der Zufuhr (oder Abfuhr) von Wärmeenergie führen. Im Mittel bewirkt daher die Reduzierung der Leistung bei Betrachtung von vielen Geräten keinen Unterschied. Außerdem ist bei Gerätetypen mit einer Wärmepumpe, die einen Kompressor haben (z.B. Kühlschrank) eine Reduzierung der aufgenommenen Leistung meist nicht möglich, da sonst der einwandfreie Betrieb gefährdet ist.

Bei Geräten der Klasse `Loadcurve_Operation` würde die aktuelle Implementierung der Funktion `Reduce_Input_Power` dazu führen, dass nur der Wert der Leistungen der einzelnen Abschnitte der Lastkurve verringert würde. Die Auswirkung dieser Reduzierung auf die Lastkurve wird aber nicht berücksichtigt, da dies einen hohen Aufwand darstellt und tiefes Wissen über die Abläufe im Gerät erfordert. Würde z.B. bei einer Waschmaschine die Heizleistung halbiert, müsste sie die Waschlauge mehr als doppelt so lange aufheizen.

Daher ist für diese beiden Geräteklassen der Einsatz dieser Output-Funktion nicht vorgesehen.

2.3.2.4. `Change_Temp_Set`

Die Geräte der Klasse `Thermal_Storage` haben eine eigene Art, die aufgenommene Leistung für einen gewissen Zeitraum zu verändern. Eine entsprechende Steuerung vorausgesetzt, könnten thermische Speicher für einen gewissen Zeitraum ihre Solltemperatur ändern. Die Höhe der Änderung wird mit dem Parameter `Temp_Set_Variation` festgelegt.

In Diagramm 2.27 sind die Auswirkungen so einer Solltemperaturänderung dargestellt. Hier wurde die Solltemperatur eines Kühlgerätes um 4°C erhöht. Deutlich ist zu Beginn der Erhöhung die relativ lange Zeitspanne ohne Aufnahme von elektrischer Leistung zu erkennen. In dieser Zeit erwärmt sich das Kühlgut im Inneren des Kühlgerätes. Dann kehrt das Gerät wieder in den normalen Betriebszustand zurück, wenn auch nun mit höherer Solltemperatur, was zu einer etwas verringerten Leistungsaufnahme führt.

Wird die Solltemperatur wieder auf das Ausgangsniveau gesetzt, reagiert das Kühlgerät mit einer längeren Kühlphase. Man könnte sagen, dass die Energie, die zuvor eingespart wurde, hier wieder aufgenommen wird, was sich aufgrund des Speicherverhaltens dieser Geräteklasse erklärt.

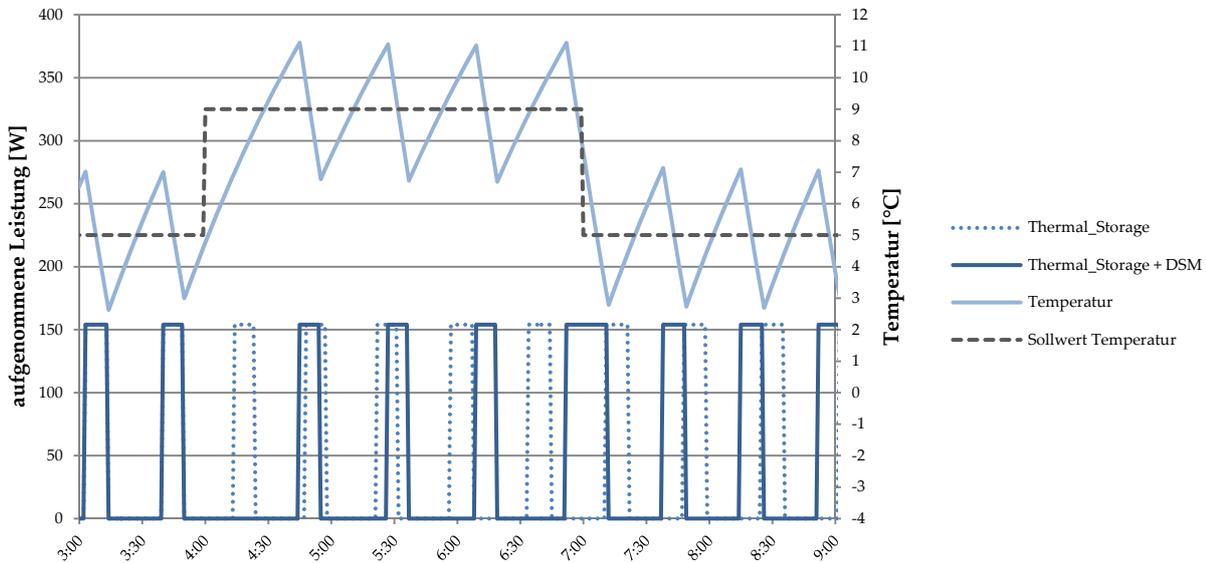


Diagramm 2.27: Change_Temp_Set bei einer Instanz von Thermal_Storage (Kühlgerät)

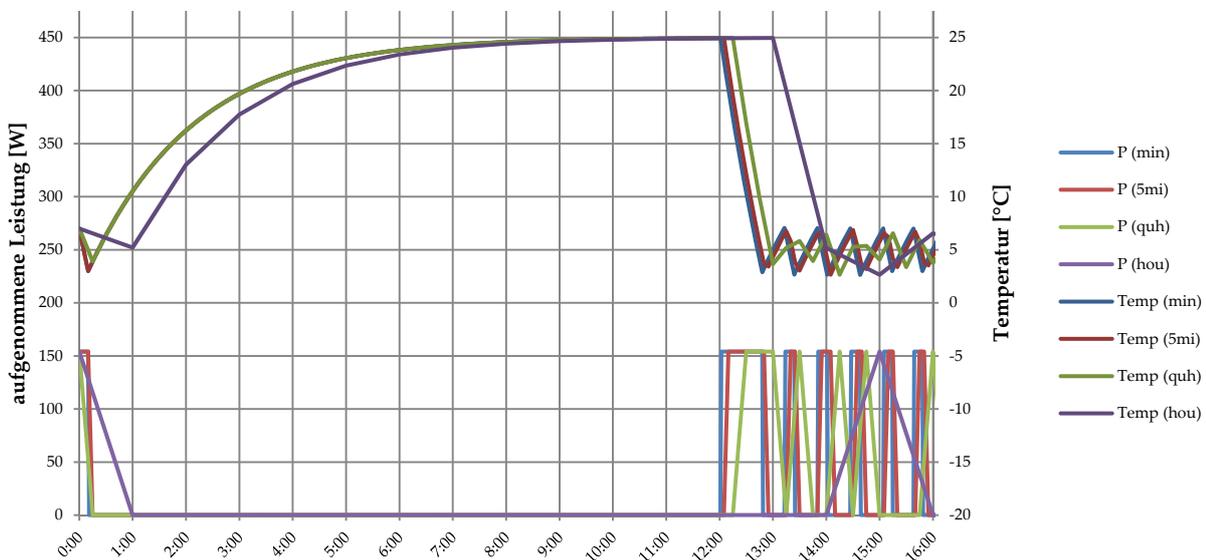


Diagramm 2.28: Auswirkung unterschiedlicher Simulationsauflösungen auf das Ergebnis bei
Check_former_Frequ_Data = 0

Wie schon im Abschnitt 2.3.1.2 bemerkt, sollte bei Geräten der Klasse Thermal_Storage immer mit den exakten Zeitwerten (Check_former_Frequ_Data = 1) und den look_back-Funktionen gearbeitet werden, da es gerade für die Temperatur im Speicher wichtig ist, wann die Solltemperatur geändert wurde.

Diagramm 2.28 stellt dar, welche Auswirkungen es hat, wenn nicht die exakten Zeitwerte von der DSM-Input-Funktion ermittelt werden. Darin wird ein Kühlgerät mittels DSM um 0:30 aus- und um 12:00 Uhr wieder eingeschaltet. Deutlich ist zu erkennen, dass das Simulationsergebnis von der gewählten Auflösung

abhängt. Je größer die Zeitabstände zwischen zwei Beobachtungszeitpunkten, desto mehr weichen die Daten voneinander ab. Bei einer Auflösung im Stunden-Raster ist die Abweichung bereits gravierend.

Der Grund dafür liegt darin, dass zwar im Hintergrund das exakte thermische Modell läuft (siehe Abschnitt 2.2.3.2), die Änderung der Solltemperatur aber in diesem Modus nur zu den Beobachtungszeitpunkten erfolgen kann. Deutlich ist das z.B. zum Zeitpunkt 12:00 Uhr zu erkennen. Statt um 12:00 beginnt das Kühlgerät bei einer Auflösung von einer Viertelstunde um erst um 12:15 wieder zu laufen, bei einer Auflösung von einer Stunde erst um 13:00. An und für sich müssten sich alle Kurven zu jeder vollen Stunde schneiden, da zu diesen Zeitpunkten immer das gleiche Ergebnis ermittelt werden sollte. Dies ist hier jedoch nicht der Fall.

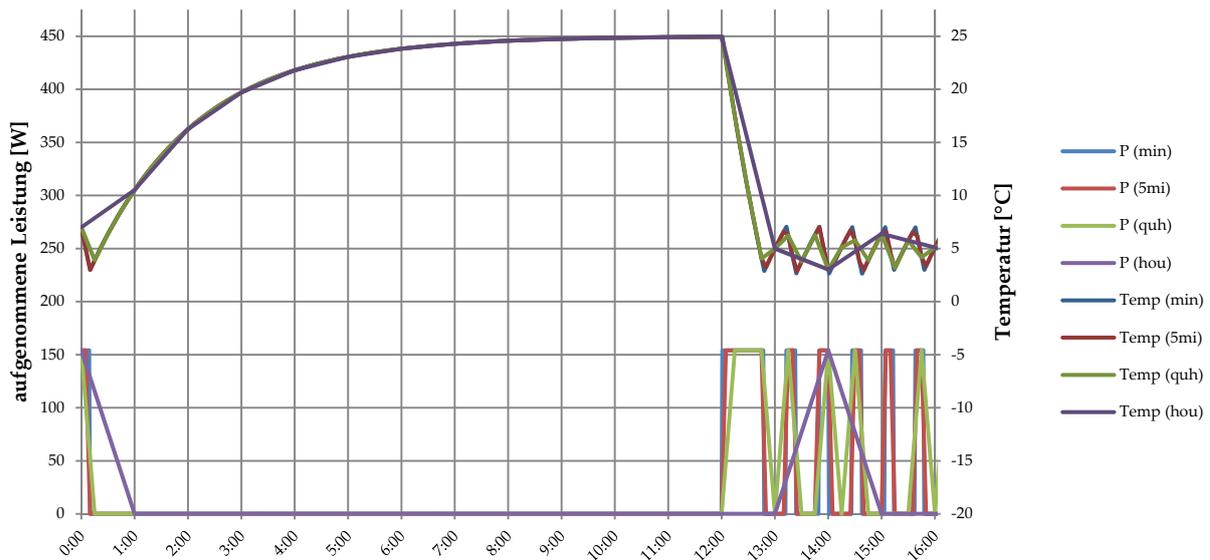


Diagramm 2.29: Auswirkung unterschiedlicher Simulationsauflösungen auf das Ergebnis bei
`Check_former_Frequ_Data = 1`

Wird in der Parameterdatei jedoch der Parameter `Check_former_Frequ_Data = 1` gesetzt, wird die `look_back`-Variante der Funktion `Change_Temp_Set` verwendet. In dieser wird mit Hilfe der durch die Input-Funktionen ermittelten Zeitpunkte `Time_Output_Start` und `Time_Output_Stop` eine exakte Ermittlung der aktuellen Werte geliefert. Z.B. ist `Time_Output_Start = 00:30` zu Beginn. Bei einer Auflösung von einer Stunde wird das thermische Modell von `Thermal_Storage` in den ersten 30 Minuten mit der alten, die nächsten 30 Minuten mit der neuen Solltemperatur berechnet. So wird zum exakt richtigen Zeitpunkt das Betriebsverhalten gewechselt und damit stimmen die Ergebnisse der unterschiedlichen Auflösungen miteinander überein. Dies ist in Diagramm 2.29 gut erkennbar, da alle Ergebnisse zur vollen Stunde gleich sind.

Damit ist gewährleistet, dass die ermittelten Ergebnisse unabhängig von der Simulationsauflösung sind.

2.3.2.5. Postpone_Start

Dieser Output-Modus kann nur von Geräten der Klasse `Loadcurve_Operation` verwendet werden, da er auf der Annahme beruht, dass das Gerät durch den Benutzer in eine Art Bereitschaftszustand versetzt werden kann, in dem dann ein Programm gestartet wird, das abgearbeitet werden soll.

Klassisches Beispiel dafür ist die Waschmaschine, die beladen wird und dann irgendwann zu Waschen beginnt. Derzeit besteht dieser Bereitschaftszustand, falls überhaupt vorhanden, bei vielen Geräten nur in einer Startzeitvorwahl, womit der Benutzer den Zeitpunkt der Fertigstellung des Waschprogramms festlegt.

In der aktuellen Implementierung der Output-Funktion `Postpone_Start` wird angenommen, dass der Benutzer die Waschmaschine gemäß seinem normalen Nutzungsverhalten verwendet, aber anstatt das Programm zu starten, die Maschine auf Bereitschaft setzt. Falls ein kritischer Netzzustand vorliegt, wird der Start des Programms verzögert.

In Diagramm 2.30 sind die Auswirkungen verschiedener Einstellungen für diesen Output-Modus zusammengefasst. Der ursprüngliche, bei der Geräteinstanzerzeugung erstellte Einsatzplan ist die punktierte Kurve. Als Parameter für die Geräteinstanz dieses Beispiels wurden die Werte aus Tabelle 2.13 genommen, jedoch alle Standardabweichungen zu Null gesetzt.

Deutlich ist zu erkennen, was mit einem bereits gestarteten Programm passiert. Dieses wird vollständig abgearbeitet, zu sehen im Verlauf links (Lastkurve 1).

Falls ein kritischer Netzzustand zum Startzeitpunkt vorliegt, und das Programm noch nicht gestartet wurde, wird der Programmstart verzögert. Mit dem Parameter `Time_Postpone_Start` wird ein mittlerer Zeitraum definiert, um den die Startzeit verschoben wird. Ist dieser Null, wird die Startzeit nur bis zur nächsten Messung der Netzfrequenz verschoben, das heißt, dass das Gerät unmittelbar, nachdem der kritische Netzzustand verlassen wird, das Programm startet, dargestellt in Diagramm 2.30 als graue Kurve.

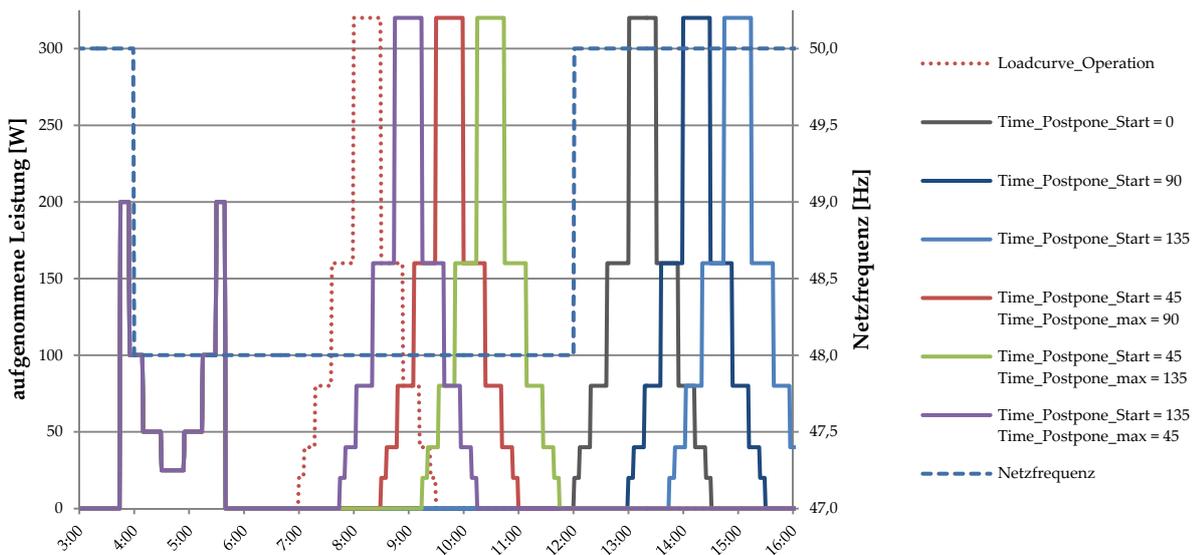


Diagramm 2.30: Darstellung der Auswirkungen verschiedener Einstellungen von `Postpone_Start` der gleichen Geräteinstanz

Ist der Parameterwert größer als Null, wird der Startzeitpunkt um die angegebene Zeitspanne verschoben. Liegt zum neuen Startzeitpunkt noch ein kritischer Netzzustand vor, wird der Startzeitpunkt neuerlich verschoben, solange, bis sich der Netzzustand normalisiert hat. Im Beispiel in Diagramm 2.30 wird der Startzeitpunkt entweder um 90 min (dunkelblaue Kurve) oder um 135 min (hellblaue Kurve) verschoben. Deutlich ist zu erkennen, dass sich die Verschiebung auch nach Erholung des Netzzustandes fortsetzt. Je nach gewählter Zeitspanne wird dann das Programm früher oder später ausgeführt. Im Beispiel wurden bei einer angegebenen Verschiebungszeitspanne von 90 min die einzelnen Startversuche um 8:30, 10:00, 11:30 und 13:00 überprüft, erst beim letzten ließ die Netzsituation ein Starten des Programms zu.

In den bisher vorgestellten Beispielen kann der Startzeitpunkt um eine beliebige Zeitspanne verschoben werden. Kommt es theoretisch zu keiner Erholung des Netzzustandes, wird das Gerät auch nie sein Programm abarbeiten können. Dies würde einen sehr starken Eingriff in die Funktionalität des Gerätes bedeuten. Auch in einem weniger dramatischen Fall könnte es dennoch zu einer unzumutbaren Verzögerung der Gerätefunktion kommen, die durch den Benutzer wahrscheinlich nicht toleriert werden würde.

Um die gesamte Zeitspanne der Verschiebung des Programmstarts zu kontrollieren, kann mit dem Parameter `Time_Postpone_max` eine Zeitspanne definiert werden, innerhalb jener das Programm gestartet werden muss. Ist der Wert Null, ergibt sich das Verhalten, das in den bisherigen Beispielen dargestellt wurde. Das heißt, es kann die Startzeit um beliebige Zeiträume verschoben werden.

Wird ein Wert größer Null definiert, ergibt sich zuerst das gleiche Verhalten der Startzeitverschiebung um die Zeitspanne `Time_Postpone_Start`. Wird jedoch die Gesamtzeit, um die der Startzeitpunkt vom ursprünglich gewünschten verschoben wurde, größer als `Time_Postpone_max`, wird das Programm ungeachtet des Netzzustandes gestartet. Der genaue Startzeitpunkt wird dabei definiert durch `Ti-`

me_Postpone_max, da dieser die Zeitspanne der maximalen Verschiebungszeit angibt. Das geht soweit, dass, falls Time_Postpone_Start größer ist als Time_Postpone_max, die Startzeit nur einmal und zwar um die Zeitspanne Time_Postpone_max verschoben wird (dargestellt in Diagramm 2.30, violette Kurve).

Erholt sich innerhalb der Zeitspanne Time_Postpone_max die Netzfrequenz, wird das Programm bereits früher, je nach vorheriger Startzeitverschiebung, gestartet.

Bei der Verwendung der Output-Funktionen Postpone_Start und Pause_Programm ist es günstig, mit Hilfe des Parameters Check_former_Frequ_Data die look_back-Versionen dieser Funktion zu verwenden, was zwar einen erhöhten Rechenaufwand und damit Rechenzeit bedeutet, bei verschiedenen Simulationsauflösungen aber zu exakten Ergebnissen führt (siehe Abschnitt 2.3.1.2).

Ein Beispiel soll diesen Effekt darstellen. Dazu wurde eine Gruppe von ca. 150 Geräten des Typs Loadcurve_Operation mit der simplen Startzeitverschiebung (ohne Time_Postpone_max) mit einer Verschiebungszeit Time_Postpone_Start = 30 Minuten ausgestattet. Der Frequenzverlauf ist in Diagramm 2.31 dargestellt. Von 3:00 weg fällt die Frequenz mit einer Rate von $\frac{5}{3}$ Hertz pro Stunde, bis sie 48 Hz erreicht. Nach einer gewissen Zeitspanne steigt die Frequenz mit der gleichen Rate, so dass sie um 15:00 das Ausgangsniveau erreicht. Alle DSM-Chips reagieren gleichzeitig bei einer Abweichung von 1 Hz, welche um 3:36 unter- und um 14:24 überschritten wird.

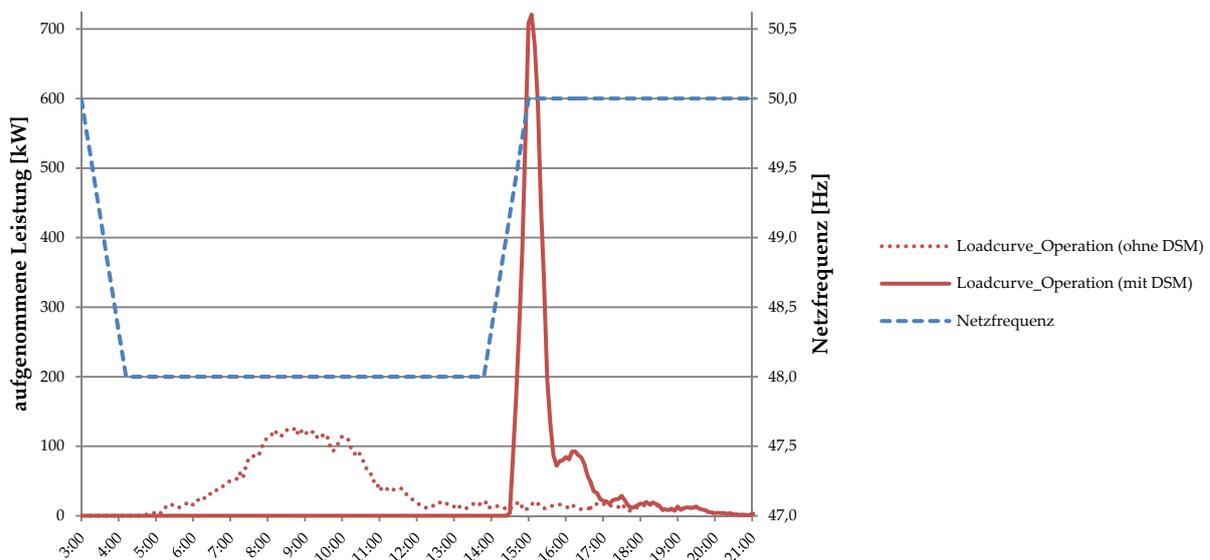


Diagramm 2.31: Ausgangssituation für Diagramm 2.32; ca. 150 Instanzen des Gerätetyps Loadcurve_Operation mit den Parametern aus Tabelle 3.19 und einer Simulationsauflösung von 5 Minuten

In Diagramm 2.31 ist gut die Auswirkung des DSM auf die Gesamtleistung zu erkennen. Alle Startzeitpunkte werden solange aufgeschoben, bis sich die Netzfrequenz erholt hat. Die ersten Geräte beginnen bereits um 14:24 mit ihren Programm. Das sind jene, deren Start ohne Beeinflussung zu den Minuten 24 und 54 jeder Stunde erfolgen würde. In der folgenden halben Stunde kommen dann alle weiteren Geräte hinzu. Da keinerlei Standardabweichungen für die DSM-Funktionen angegeben wurden, kommt ein, aus der Sicht des Netzes, denkbar ungünstiger Verlauf bei der Rückkehr der Frequenz auf den Normalpegel zustande. Statt über die normale Funktionszeit verteilt beziehen fast alle Geräte beinahe gleichzeitig Energie, was sich in einer dementsprechenden Leistungsspitze widerspiegelt.

Diese Lastspitze wird in Diagramm 2.32 genauer untersucht. Darin werden unterschiedliche Simulationsauflösungen mit und ohne die look_back-Versionen der Input- und Output-Funktionen verglichen.

Wurde die Option Check_former_Frequ_Data nicht aktiviert, kommt es zu deutlichen Abweichung bei den unterschiedlichen Simulationsauflösungen, besonders gut ersichtlich um 15:00 Uhr (rote Kurven). Diese kommen dadurch zu Stande, weil die Frequenzmessung und die Verschiebung der Startzeiten hier nur zu den Beobachtungszeitpunkten durchgeführt wird, welche von der Simulationsauflösung abhängen. Gerade der Zeitpunkt der Erholung der Netzfrequenz (14:24 Uhr) wird hier durch die Beobachtungszeit-

punkte nie genau getroffen. So kommt es dazu, dass z.B. bei der Stundenauflösung in der Zeit von 14:00 bis 15:00 keine Startzeitverschiebung mehr stattfindet und so innerhalb dieser Stunde bereits alle Geräte mit ihrem Programm beginnen, was aber aufgrund der Simulationsauflösung nicht dargestellt werden kann. Anders gesagt bedeutet eine gröbere Auflösung eine Verschiebung der Lastspitze hin zu früheren Zeitpunkten. Statt um 14:24 Uhr reagieren die Geräte z.B. schon um 14:00, was bereits eine hohe Abweichung darstellt.

Durch die Berücksichtigung der genauen Zeitpunkte sind die Simulationsergebnisse unabhängig von der Simulationsauflösung (grüne Kurven).

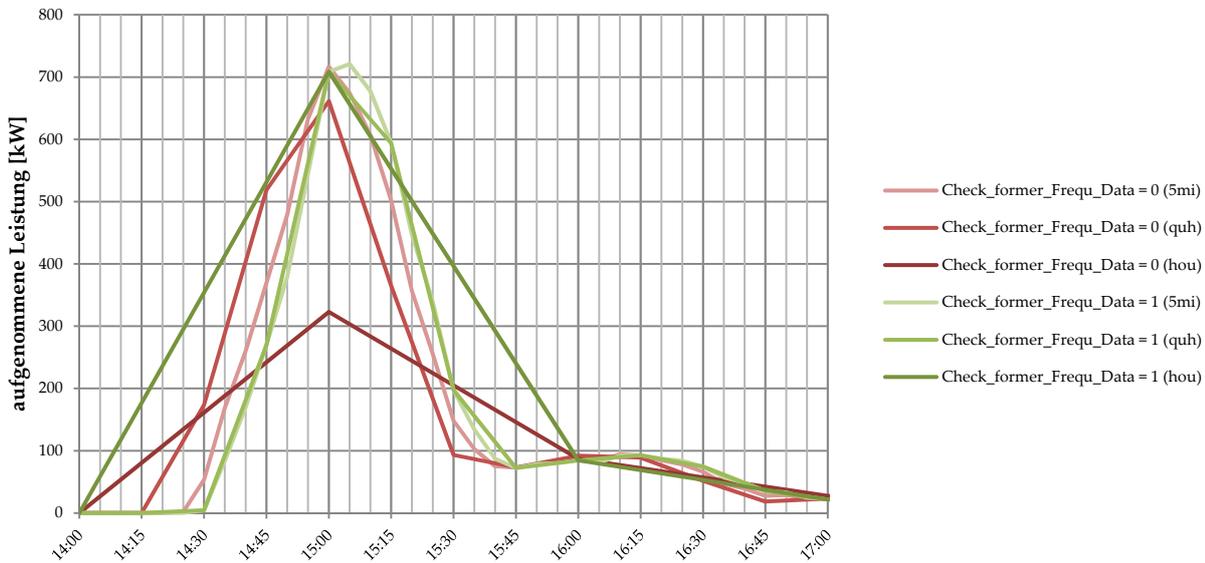


Diagramm 2.32: Auswirkung unterschiedlicher Simulationsauflösungen auf das Simulationsergebnis

2.3.2.6. Pause_Programm

In der Output-Funktion `Postpone_Start` wird ein bereits aktives Programm nicht unterbrochen, wenn sich ein kritischer Netzzustand einstellt. Falls man jedoch von einer höheren Kooperation eines Gerätes des Typs `Loadcurve_Operation` ausgeht, könnte dieses ein bereits laufendes Programm unterbrechen, um so das Netz unmittelbar zu entlasten. Dies setzt aber voraus, dass das Gerät diese Unterbrechung zulassen bzw. das Programm fortsetzen kann.

Bereits in Abschnitt 2.2.7.1 wurde erläutert, dass für die Lastkurven der Geräteklasse `Loadcurve_Operation` Teilbereiche definiert werden können, die in ihrem Ablauf nicht unterbrochen werden sollten. Dieses Wissen besitzt die Steuerung des jeweiligen Gerätes, die dahingehend reagiert.

Mit der Option `Consider_non_stop_parts` kann festgelegt werden, ob das Gerät nun Rücksicht auf diese Teile der Lastkurve nimmt, oder nicht.

In Diagramm 2.33 sind die Unterschiede dargestellt. Es wurde die gleiche Geräteinstanz verwendet, die auch Grundlage für Diagramm 2.30 war. Die punktierte Linie stellt den ursprünglichen Einsatzplan dar, zu dem es kommen würde, wenn kein DSM durchgeführt werden würde. Als nicht unterbrechbare Bereiche der Lastkurve wurden jene Teile definiert, welche die drei höchsten Leistungsbereiche enthalten (siehe Tabelle 2.13).

Bei der roten Kurve wird dieser Teil der Lastkurve zuerst abgearbeitet und der verbleibende Programmabschnitt erst nach der Erholung der Netzfrequenz fortgesetzt.

Im Fall der grünen Kurve wird das Programm sofort unterbrochen. Dieser nicht unterbrechbare Teil der Lastkurve wird aber, wenn das Gerät wieder sein Programm fortsetzt, noch einmal abgearbeitet. Wie in Abschnitt 2.3.2.1 beim Output-Modus `Turn_Off` von Geräten der Klasse `Loadcurve_Operation` angemerkt, soll dies z.B. das Abkühlen und Wiederaufheizen der Waschlauge simulieren. Dies ist aber eine sehr einfache Näherung, die insbesondere bei kurzen Unterbrechungen ihre Gültigkeit verlieren kann, da z.B.

die Waschlauge nicht so schnell abkühlt. Diese Näherung stellt aber den „Worst-Case“ dar, weshalb sie beibehalten wurde.

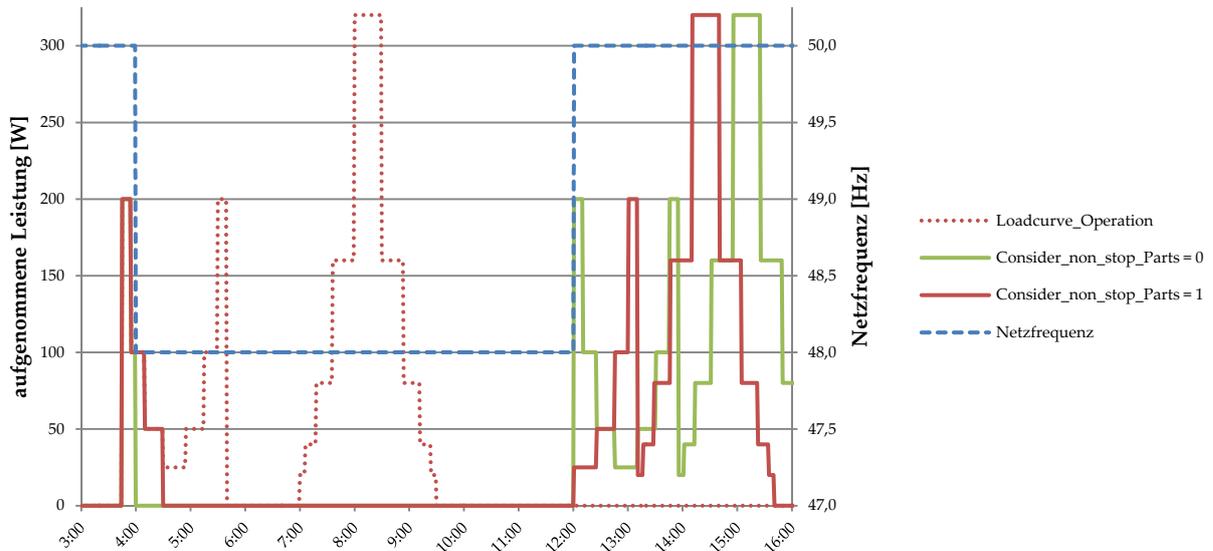


Diagramm 2.33: Einfluss der Option `Consider_non_stop_Parts`

Weiters ist in Diagramm 2.33 zu erkennen, was passiert, wenn ein Gerät mehrmals starten würde, sich die Startzeiten aber aufgrund der DSM-Funktion verschieben. Es wird gleich vorgegangen wie bei der Erstellung des Einsatzplanes für die Geräteklasse `Loadcurve_Operation` (siehe Abschnitt 2.2.7.2). Um zu verhindern, dass sich einzelne Lastkurven überlappen, werden die jeweiligen Startzeiten so angepasst, dass die einzelnen Programme hintereinander ausgeführt werden.

Im Abschnitt 2.2.7.2 wurde schon auf das Problem bei dieser Vereinfachung eingegangen, die an und für sich nicht das reale Nutzerverhalten nachbildet. Da aber der Fall, dass mehrere Programme am selben Tag gestartet werden, eher selten vorkommt, wird auf die Behandlung dieser Sonderfälle verzichtet.

Das Pausieren der Programmausführung wird so bewerkstelligt, dass eine unterbrochene Lastkurve in zwei Teile gespaltet wird. Der erste Teil ist der bereits absolvierte, der zweite der noch zu absolvierende Teil. Im Fall der Unterbrechung eines nicht unterbrechbaren Bereichs wird der betroffene Bereich noch einmal komplett vor den noch zu absolvierenden Teil gesetzt.

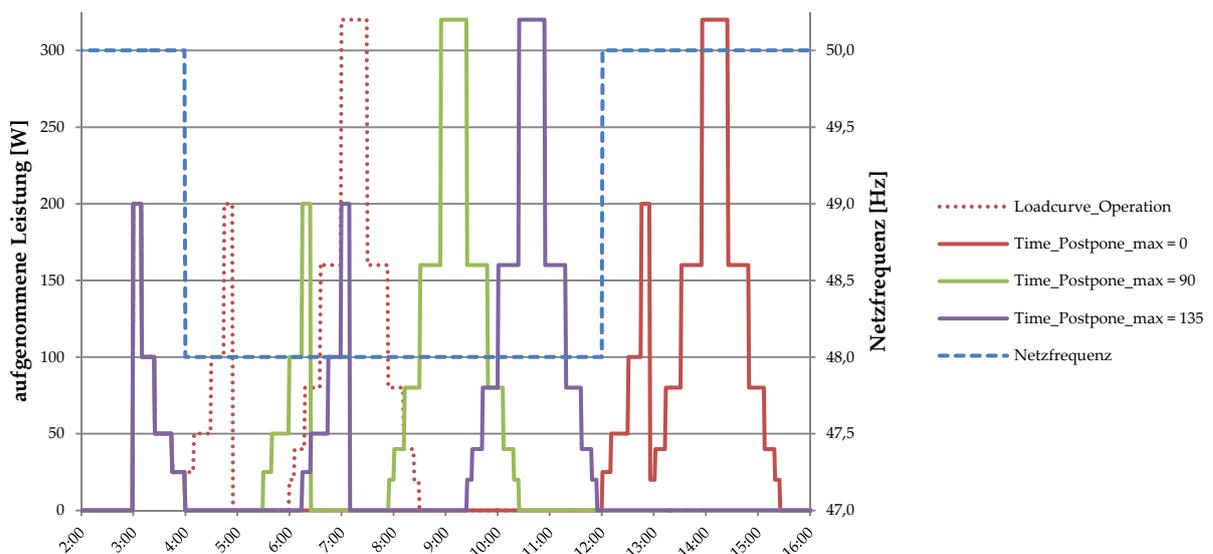


Diagramm 2.34: Einfluss des Parameters `Time_Postpone_max` auf die Output-Funktion `Pause_Programm`

Mit den Teillastkurven wird nun genauso eine Startzeitverschiebung durchgeführt wie beim Output-Modus `Postpone_Start`, weshalb auch für `Pause_Programm` die gleichen Parameter (`Ti-`

me_Postpone_Start, Time_Postpone_max und Check_former_Frequ_Data) in gleicher Weise definiert werden können.

In Diagramm 2.34 sind drei verschiedenen Werte für Time_Postpone_max angenommen worden. Weiters wird die erste Lastkurve hier in einem unterbrechbaren Teil pausiert, wodurch der Parameter Consider_non_stop_Parts keine Auswirkungen auf das Ergebnis hätte.

Durch die Definition einer maximalen Verschiebungszeit wird gewährleistet, dass das Gerät in einer definierten Zeitspanne seinen Programmablauf beendet. Kommt es aufgrund von mehreren Starts des Gerätes zu aufeinanderfolgenden Lastkurven, wird die nächste Lastkurve ebenfalls verschoben, maximal jedoch um die angegebene Zeitspanne Time_Postpone_max (z.B. grüne oder violette Kurve in Diagramm 2.34).

Der Output-Modus Pause_Programm stellt jene DSM-Funktionalität dar, die besonders tief in das Geräteverhalten eingreifen muss. Ob dieser Anwendungsfall daher in absehbarer Zeit umsetzbar wäre, ist eher fragwürdig, da hier eine sehr gute Integration von der Geräteseite erforderlich ist. Dennoch wird in den Simulationsergebnissen auf die Auswirkungen dieses Geräteverhaltens genauer eingegangen.

2.4. Frequenzdaten

Für die Beeinflussung der Geräteleistung soll bei den zu untersuchenden DSM-Systemen vordergründig die Netzfrequenz die Notwendigkeit für einen Eingriff liefern.

Die Netzfrequenz wird im Zuge der Simulation als eine von außen vorgegebene Größe betrachtet, die durch das Verbraucherverhalten nicht beeinflusst wird. Diese Annahme spiegelt wieder, dass die Leistung der betrachteten Verbraucher klein gegenüber der im Netz verfügbaren ist. Eine zweite Möglichkeit könnte auch eine innerhalb eines Inselnetzes installierte Regelung sein, die mit Hilfe der Netzfrequenz das Verbraucherverhalten zu steuern versucht.

Mit Hilfe der Funktion create_frequency_data lassen sich verschiedene Frequenzverläufe generieren. In dieser Funktion sind Hilfsfunktionen implementiert, mit deren Hilfe Rampen, konstante Verläufe und Rauschen der Frequenz erzeugt werden können. Für zukünftige Erweiterungen der Software wäre auch das direkte Einlesen von gemessenen Frequenzdaten vorgesehen.

Die generierten Frequenzdaten liegen zumindest in 30 Sekunden Schritten vor und werden in der Funktion create_frequency_data an die Simulationslaufzeit angepasst. Die Frequenzdaten werden als $[2, n]$ -Matrix im Simulationsprogramm verarbeitet, wobei die erste Zeile die Zeitpunkte der jeweiligen Frequenzwerte enthält, die in der zweiten Zeile aufgelistet sind. Im Simulationsprogramm werden dann den einzelnen DSM_Device-Instanzen alle Frequenzdaten, die für den aktuellen Simulationsschritt notwendig sind, übergeben (siehe Abschnitt 2.5.3).

Mit Hilfe des GUI können die aktuellen Frequenzdaten betrachtet, neue generiert und diese gespeichert bzw. geladen werden (Menü Frequenzdaten). Die Frequenzdatendateien werden durch die Endung „.fqd“ besonders gekennzeichnet.

2.5. Simulationsdurchführung und Datenausgabe

Mit den entwickelten Klassen für die Geräte und die DSM-Einheiten kann nun ein Gerätekollektiv gebildet und damit ein DSM-System simuliert werden.

Im Folgenden sollen kurz die Abläufe erläutert werden, welche bei der Simulation stattfinden und wie die einzelnen Klassen hierbei zusammenwirken.

2.5.1. Erzeugung des Gerätekollektivs

Abbildung 2.4 zeigt schematisch, wie die einzelnen Instanzen vor der Simulation erstellt werden. Aus der Parameterdatei werden für jeden Gerätetyp die jeweiligen Parameterwerte ausgelesen (siehe Abschnitt 2.1). Jeder Gerätetyp wird mit jener Geräteklasse modelliert, die das Verhalten dieser Geräte am besten nachbildet. Bei den Geräteinstanzen werden die Parameter zuerst einer Streuung unterzogen um für jede Instanz einen individuellen Parametersatz zu erhalten (siehe Abschnitt 2.1.2). Ähnlich wird hier auch bei den DSM_Device-Instanzen vorgegangen, nur dass hier zusätzlich auch eine Auswahl der passenden Input- und Output-Funktionen stattfindet (siehe Abschnitt 2.3).

Nachdem die Geräte- und `DSM_Device`-Instanzen erzeugt wurden, werden diese für jedes einzelne Gerät miteinander verkoppelt. Dazu wird die für das jeweilige Gerät erzeugte `DSM_Device`-Instanz dem Attribut `DSM` der Geräteinstanz zugeordnet. Man könnte sagen, die DSM-Einheit wird in das Gerät „eingebaut“. In Abbildung 2.4 ist das durch die strichlierte Verbindung der Instanzen angedeutet.

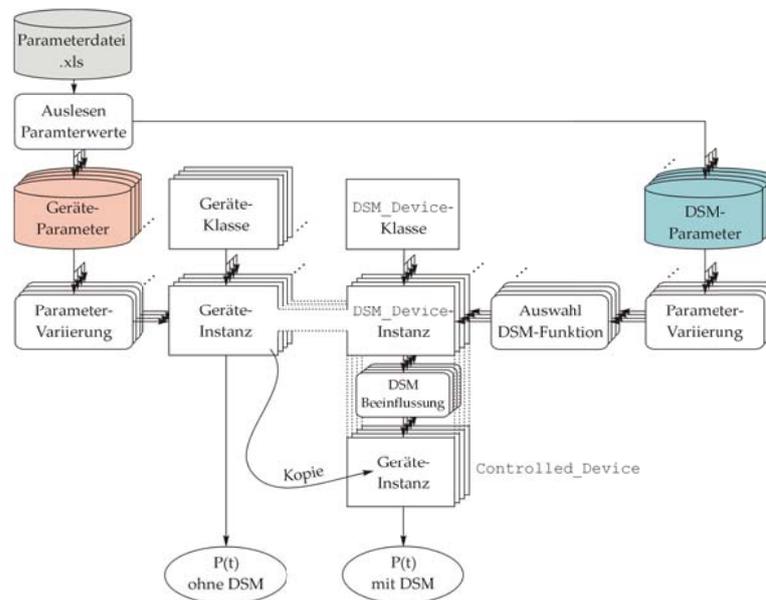


Abbildung 2.4: Schema der Erzeugung der Geräteinstanzen und der `DSM_Device`-Instanzen

Damit beim Simulationdurchlauf das Gerätekollektiv mit und ohne Einsatz von DSM untersucht werden kann, wird der `DSM_Device`-Instanz eine Kopie der Geräteinstanz zur Verfügung gestellt (`Controlled_Device`), das die DSM-Einheit beeinflusst. Die ursprüngliche Geräteinstanz ist von diesen Beeinflussungen nicht betroffen.

2.5.2. Simulationsablauf

Nach dem Start der Simulation werden alle Parameter aus der angegebenen Parameterdatei eingelesen. Falls dies angegeben wurde (siehe Abschnitt 2.6.2), werden auch die Simulationseinstellungen aus der Parameterdatei übernommen. Danach werden, wiederum abhängig von den getroffenen Einstellungen, entweder die Frequenzdaten geladen oder neu generiert.

Nachdem die Geräte- und `DSM_Device`-Instanzen erzeugt wurden, wird nun für jeden Simulationszeitschritt und für jedes Gerät dessen Reaktion ermittelt. In den meisten Fällen bestehen die ermittelten Daten aus der zum aktuellen Simulationszeitpunkt aufgenommenen Leistung des Gerätes (siehe Abbildung 2.5).

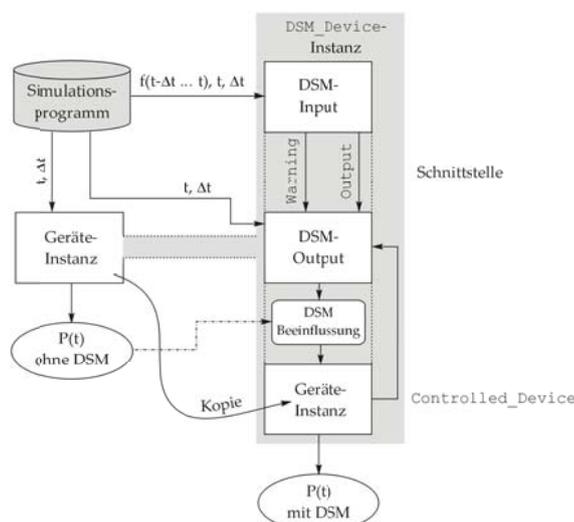


Abbildung 2.5: Aufbau der `DSM_Device`-Instanz sowie Ablauf der Ermittlung der aufgenommenen Leistung für einen Simulationsschritt (ohne `look_back`-Funktion)

Dazu stellt das Simulationsprogramm jeder Instanz im aktuellen Simulationsschritt alle, für die Ermittlung der Reaktion notwendigen Informationen zur Verfügung. Zuerst wird das unbeeinflusste Gerät simuliert. Dieses benötigt dafür den aktuellen Zeitpunkt t (Datum & Uhrzeit) sowie die Schrittweite der Simulation Δt . Daraus wird mit Hilfe der Funktion `next_step` die aktuelle Leistungsaufnahme ermittelt.

Danach wird das Gerät mit DSM simuliert. In Abbildung 2.5 ist auch der in Abschnitt 2.3 dargestellte Aufbau der `DSM_Device`-Instanz zu erkennen. Dem DSM-Input werden neben den aktuellen Zeitpunkt und die Simulationsschrittweite auch alle Frequenzdaten des letzten Simulationsschrittes $f(t - \Delta t \dots t)$ zur Verfügung gestellt. Aus diesen Daten werden einerseits die Informationen für die Schnittstelle (Attribute `Warning` und `Output`) ermittelt. Andererseits werden auch, falls dies notwendig ist, mit den `look_back`-Funktionen die genauen Zeitpunkte der Leistungsbeeinflussung berechnet (siehe Abschnitt 2.3.1.2.3).

Über die Schnittstelle wird der DSM-Output angesteuert, der die Beeinflussung der Kopie der Geräteinstanz je nach Anforderung veranlasst und mit dieser die aktuelle Leistungsaufnahme ermittelt. Oft kann dafür bereits das Ergebnis der ursprünglichen Geräteinstanz verwendet und damit der Simulationaufwand verringert werden.

Die so ermittelten Leistungswerte werden zu den bisher zu diesem Zeitpunkt ermittelten Leistungen aller Geräte des gleichen Typs addiert, jeweils mit und ohne Einsatz von DSM. Nach der Simulation liegen zwei $[n, m]$ -Matrizen vor, die für die m Simulationszeitpunkte die aktuelle Leistungsaufnahme der n Geräteklassen enthält.

Das Ergebnis wird nach der Simulation zu einer MATLAB-Struktur zusammengefasst, die später noch um zusätzliche Daten erweitert wird (siehe Abschnitt 2.5.5).

2.5.3. Einzelsimulation

Bei einer Einzelsimulation (Standardeinstellung) wird ein kompletter Simulationsdurchlauf gemäß den zuvor definierten Einstellungen durchgeführt.

Sind vor dem Start der Simulation bereits Instanzen von Geräten oder DSM-Einheiten vorhanden, überprüft das Programm, ob diese für die aktuelle Simulation weiterverwendet werden könnten. Ist dies der Fall, erfolgt eine Frage an den Benutzer, ob er eine Weiterverwendung der aktuellen Instanzen wünscht. Dabei hat dieser bis zu drei verschiedene Möglichkeiten:

- Verwerfen sowohl der Geräte- als auch der `DSM_Device`-Instanzen und erzeugen neuer Instanzen. Diese Option sollte speziell dann gewählt werden, wenn Parameter geändert wurden, damit diese Änderungen berücksichtigt werden.
- Verwerfen der `DSM_Device`-Instanzen und weiterverwenden der Geräteinstanzen. Es werden für die vorhandenen Geräte neue `DSM_Device`-Instanzen erzeugt. Diese Option sollte speziell dann gewählt werden, wenn die DSM-Parameter geändert wurden.
- Weiterverwenden sowohl der Geräte als auch der `DSM_Device`-Instanzen. So wird wieder die exakt gleiche Gerätekonstellation simuliert. Diese Option sollte speziell dann gewählt werden, wenn nur Simulationseinstellungen geändert wurden und die gleichen Geräte verwendet werden sollen.

2.5.4. Simulationsreihe

Um mehrere Simulationsdurchläufe automatisch hintereinander durchführen zu können, besteht die Möglichkeit, Simulationsreihen zu definieren.

Für jeden einzelnen Simulationsdurchlauf werden verschiedene Quellen für die Parameter und optional den Frequenzdaten in einer sogenannten „Job-Liste“ festgelegt. Mit Hilfe des GUI kann diese Job-Liste erstellt und gespeichert werden (siehe Abschnitt 2.6.2). Diese liegt dann als EXCEL-Datei vor und lässt sich ähnlich wie die Parameterdateien somit einfach betrachten und verwalten.

Tabelle 2.19 gibt ein Beispiel für eine definierte Job-Liste. Neben den Pfad zur Parameterdatei des jeweiligen Simulationsdurchlaufes wird zusätzlich ein Link zur Verfügung gestellt, mit dem einfach die jeweilige

Parameterdatei überprüft bzw. bearbeitet werden kann. Weiters wurden im gezeigten Beispiel noch unterschiedliche Frequenzdaten für jeden der Simulationsdurchläufe angegeben. Diese müssen aber zu den jeweiligen Simulationseinstellungen angepasst sein (siehe Abschnitt 2.4).

Ob die Simulationseinstellungen aus der Parameterdatei geladen werden sollen und wie mit bereits vorhandenen Geräte- und `DSM_Device`-Instanzen verfahren werden soll, kann vor dem Start der Simulationsreihe festgelegt werden (siehe Abschnitt 2.6.2).

Auflistung der Parameterdateien für Simulationsdurchlauf:									
Lfd. Nr.	Link zu Datei	Pfad der Datei							
1	LINK	D:\Verbrauchersimulation mit DSM\Default_Parameterwerte.xls							
	Frequ_Data:	D:\Verbrauchersimulation mit DSM\Frequenzdaten\Test_Parameterstreuung_DSM_Input.fqd							
2	LINK	D:\Verbrauchersimulation mit DSM\Messreihen\Messreihe_5\Messdurchlauf_1.xls							
	Frequ_Data:	D:\Verbrauchersimulation mit DSM\Frequenzdaten\Reduzierung_Lastspitzen_ausgewählte_Daten.fqd							

Tabelle 2.19: Beispiel für eine Job-Listendatei in der neben unterschiedlichen Parameterdateien auch unterschiedliche Frequenzdaten definiert werden.

2.5.5. Nachbearbeitung der Daten

Nach jedem Simulationsdurchlauf werden die ermittelten Simulationsdaten einer Nachbereitung zugeführt. Diese erfolgt in der Funktion `calculate_infos`. In dieser wird ermittelt, wie viele Geräte im Einsatz waren, der Durchschnittsverbrauch jeder Geräteklasse sowie die Gesamtleistungsaufnahme.

Zusätzlich werden noch die Simulationszeitpunkte in der Ergebnisstruktur festgehalten.

In dieser Funktion besteht die Möglichkeit noch weitere Daten aus dem Simulationsergebnis zu ermitteln, sofern dies benötigt wird.

2.5.6. Speichern der Daten

Während und nach der Simulation werden laufend die Simulationsergebnisse gespeichert. Mit Hilfe des GUI kann eine Datei definiert werden, in die sämtliche erzeugte Dateien abgelegt werden sollen (siehe Abschnitt 2.6.1.1).

In diesem Verzeichnis wird immer ein Unterordner mit dem aktuellen Simulationsdatum erstellt, in dem dann die Simulationsergebnisse zu finden sind. Bei einer Simulationsreihe wird nach dem Datum im Ordnernamen auch der Name der abgearbeiteten Job-Liste angeführt.

Bei jedem Simulationsdurchlauf werden automatisch bis zu 5 verschiedene Dateien erzeugt und gespeichert. Um diese verschiedenen Dateien in Zusammenhang zu bringen, werden alle Namen vom Zeitpunkt des Starts des Simulationsdurchlaufes angeführt (siehe Abbildung 2.6). Als zusätzliche kurze Information wird am Ende jedes Dateinamen noch die zeitliche Auflösung der Simulation sowie die Anzahl der Personen angeführt, um sich schneller zurechtzufinden.

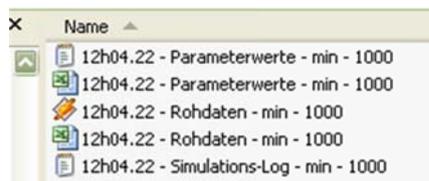


Abbildung 2.6: erzeugte Dateien nach einem Simulationsdurchlauf

Die Parameterwerte werden einmal als `.txt`-Datei und ein zweites Mal als EXCEL-Arbeitsmappe (`.xls`) gespeichert. Diese beiden Dateien enthalten alle Informationen über die gemachten Einstellungen und Parameterwerte, um dieselbe Simulation wiederholen zu können. Die `.xls`-Parameterdatei kann vom Simulationsprogramm direkt eingelesen werden (siehe Abschnitt 2.1).

Die Ergebnisse der Simulation werden als MATLAB-Datendatei (`.mat`) und optional als EXCEL-Arbeitsmappe (`.xls`) gespeichert („Rohdaten“). In der `.mat`-Datei sind sämtliche Ergebnisse sowie alle Einstellungen, Frequenzdaten und Geräteinstanzen als MATLAB-Strukturen gespeichert, die vom Simulationsprogramm geladen und weiterverarbeitet werden können. So können z.B. die Geräteinstanzen einer bestimmten Simulati-

on für einen neuen Simulationsdurchlauf herangezogen werden. Aus dieser Datei kann auch später, falls dies nicht automatisch nach der Simulation gemacht worden ist, das Ergebnis als .xls-Datei gespeichert werden (siehe Abschnitt 2.6.1.1).

In der .xls-Datei wird neben den Simulationszeitpunkten auch die Leistungsaufnahme (in kW) der einzelnen Gerätetypen sowohl mit und ohne DSM-Einsatz, die Gesamtleistung (in kW), der Durchschnittsverbrauch (in W) sowie die Anzahl der aktiven Geräte gespeichert. Die Zeiten werden beim Speichern von den seriellen MATLAB-Zeit-Zahlen in die entsprechenden EXCEL-Zeit-Zahlen umgerechnet, um in EXCEL direkt mit den Daten weiterarbeiten zu können.

Das Simulations-Log wird während der Simulation erstellt und enthält die Ausgaben der MATLAB-Konsole, die während des Durchlaufs gemacht werden. Weiters gibt es Aufschluss über die Dauer eines Simulationsdurchlaufs und eventuelle Fehler, die während der Simulation aufgetreten sind.

Bei einer Simulationsreihe werden die einzelnen Dateien immer dem Zeitpunkt des Startes des jeweiligen Durchlaufes zugeordnet. Auch wird bei einer Simulationsreihe nur ein Simulationslog erstellt, in dem die Ausgaben der einzelnen Durchläufe zusammengefasst sind.

2.5.7. Anzeigen der Daten

Nach der Simulation können die Daten direkt mit dem Simulationsprogramm visualisiert werden. Abbildung 2.7 zeigt ein Beispiel so einer Ausgabe. Neben der aufgenommenen Leistung ohne DSM (punktiert dargestellt) und mit DSM (durchgezogen) der einzelnen Gerätegruppen und der Gesamtleistung wird auch die Netzfrequenz zur Darstellung gebracht.

Auch können frühere Simulationsdaten in das Programm geladen und zur Anzeige gebracht werden (siehe Abschnitt 2.6.1). In der Titelzeile wird zusätzlich das Datum und die Uhrzeit des dargestellten Ergebnisses angezeigt.

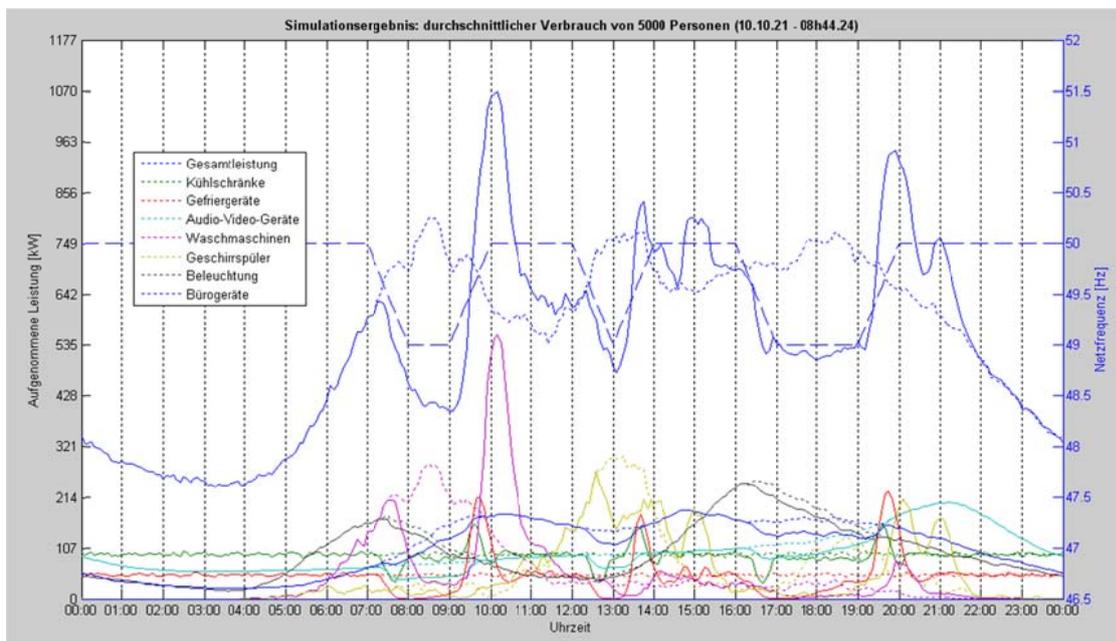


Abbildung 2.7: Beispiel für eine Ausgabe der Simulationsergebnisse durch das Simulationsprogramm

2.6. Programmoberfläche - GUI

Für die einfache Bedienung des Simulationsprogrammes wurde ein *Graphical User Interface* (GUI) programmiert, das es erlaubt, die wichtigsten Einstellungen direkt vorzunehmen und das während der Simulation den Fortschritt dieser anzeigt.

Abbildung 2.8 zeigt die grobe Datenstruktur des Simulationsprogrammes, und wie der Benutzer dieses verwendet. Sowohl die wichtigsten Eigenschaften als auch der Umgang mit der Parameterdatei, der Job-Liste, den Frequenzdaten und den Simulationseinstellungen wurde bereits genauer erörtert.

Die Software bietet zusätzlich die Möglichkeit, bereits vorhandene Simulationsdaten zu laden (Sim.-Daten) um diese darzustellen oder auch die darin enthaltenen Geräteinstanzen für eine neue Simulation heranzuziehen.

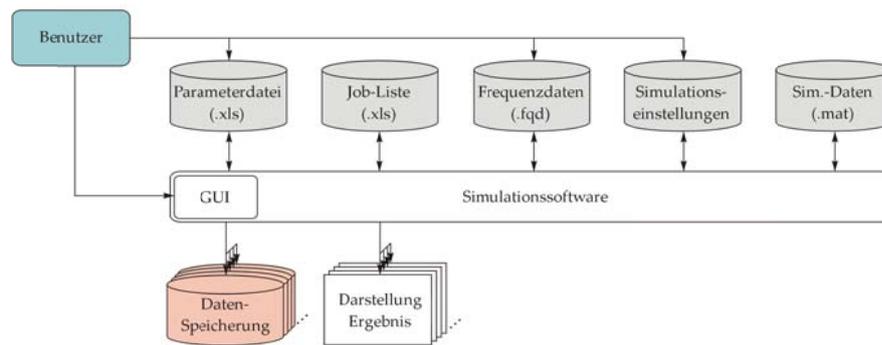


Abbildung 2.8: Schema des Aufbaus der Simulationssoftware

Im Folgenden soll kurz auf die Bedienung der Software eingegangen und die wichtigsten Funktionen kurz erläutert werden.

2.6.1. Hauptfenster

Abbildung 2.9 zeigt das Hauptfenster des Simulationsprogrammes. Darin lassen sich die wichtigsten Simulationseinstellungen direkt vornehmen (Startzeitpunkt, Endzeitpunkt, Anzahl Personen, Zeitliche Auflösung, Zusammenstellung Verbraucher, DSM aktivieren; siehe Abschnitt 2.1.1).

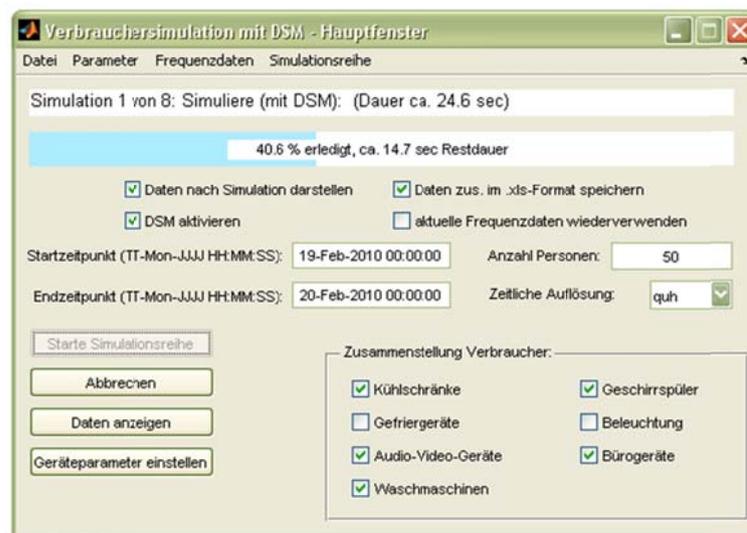


Abbildung 2.9: Hauptfenster des Simulationsprogrammes

Während der Simulation wird über eine Statuszeile und einen Fortschrittsbalken der Simulationsfortschritt angezeigt. Die weiteren Optionen sind:

Daten nach Simulation darstellen

Ist diese Option ausgewählt, wird nach jedem Simulationsdurchlauf das Ergebnis der Simulation angezeigt (siehe Abschnitt 2.5.7). Bei einer Simulationsreihe wird für jeden Durchlauf ein neues Fenster geöffnet.

Daten zus. im .xls-Format speichern

Über diese Option kann angegeben werden, ob die Simulationsdaten nach der Simulation zusätzlich neben der .mat-Datendatei auch als EXCEL-Arbeitsmappe gespeichert werden sollen (siehe Abschnitt 2.5.6).

aktuelle Frequenzdaten wiederverwenden

Über diese Option kann angegeben werden, ob die aktuellen Frequenzdaten für die Simulation herangezogen werden sollen (können über Menü Frequenzdaten → aktuelle Frequenzdaten anzeigen... dargestellt werden), oder ob vor jedem Simulationsdurchlauf neue Frequenzdaten mit Hilfe der Funktion

`create_frequency_data` generiert werden sollen (siehe Abschnitt 2.4). Diese Option sollte aktiv sein, falls Frequenzdaten aus einer Datei geladen wurden. Sind keine gültigen Frequenzdaten vorhanden, ist diese Option ausgegraut.

Starte Simulation bzw. Starte Simulationsreihe

Mit dieser Schaltfläche wird die Simulation mit den vorher definierten Einstellungen gestartet. Über den Button „Abbrechen“ kann eine laufende Simulation jederzeit unterbrochen werden.

Daten anzeigen

Über diese Schaltfläche können, sofern vorhanden, die aktuellen Simulationsdaten betrachtet werden (siehe Abschnitt 2.5.7).

Geräteparameter einstellen

Diese Schaltfläche öffnet die aktive Parameterdatei, um diese zu bearbeiten (siehe Abschnitt 2.1). Wichtig ist das Speichern dieser Parameterdatei, nachdem die gewünschten Änderungen vorgenommen wurden, damit diese auch vom Simulationsprogramm übernommen werden können.

2.6.1.1. Menü Datei

Daten laden...

Lädt zuvor gespeicherte Simulationsdaten einer .mat-Datendatei. Neben dem Simulationsergebnis werden die für die Simulation verwendeten Geräteinstanzen, Frequenzdaten und Einstellungen geladen und können für die folgende Simulation weiterverwendet werden. Auch kann das geladene Simulationsergebnis betrachtet werden (Schaltfläche „Daten anzeigen“).

Daten speichern

Die aktuellen Daten werden gespeichert (siehe Abschnitt 2.5.6).

Daten im .xls-Format speichern

Das aktuelle Simulationsergebnis wird als EXCEL-Arbeitsmappe gespeichert. Ist diese Datei bereits vorhanden, ist diese Option ausgegraut.

Speicherort für Daten...

Legt den Speicherort für die Simulationsergebnisse fest (siehe Abschnitt 2.5.6).

Beenden

Das Programm beenden.

2.6.1.2. Menü Parameter

Geräteparameter laden...

Festlegen einer Parameterdatei. Wurde eine .xls-Datei ausgewählt, wird diese überprüft. Ist diese keine Parameterdatei kommt es zu einer Fehlermeldung.

aktuelle Geräteparameter speichern unter...

Öffnet einen Dialog um die aktuellen Geräteparameter zu speichern.

Geräteparameter einstellen...

Öffnen der aktuellen Parameterdatei um diese zu bearbeiten.

Geräteparameter anzeigen

Öffnen der aktuellen Parameterdatei.

2.6.1.3. Menü Frequenzdaten

aktuelle Daten anzeigen...

Zeigt die aktuell verfügbaren Frequenzdaten an. Sind keine Frequenzdaten vorhanden, ist diese Option ausgegraut.

Daten laden...

Laden von Frequenzdaten.

Daten speichern unter...

Speichern der aktuellen Frequenzdaten als .fqd-Datei.

Daten generieren

Ausführen der Funktion `create_frequency_data`, um Frequenzdaten zu den aktuellen Simulationseinstellungen zu erzeugen.

Generierung einstellen...

Öffnen der .m-Datei der Funktion `create_frequency_data` um darin festzulegen, wie die Frequenzdaten erzeugt werden sollen.

2.6.1.4. Menü Simulationsreihe

Job-Liste laden...

Laden einer Job-Liste für eine Simulationsreihe (siehe Abschnitt 2.5.4).

Einstellungen...

Fenster für die Einstellungen für eine Simulationsreihe öffnen (siehe Abbildung 2.10 und folgender Abschnitt).

2.6.2. Einstellungen für Simulationsreihen

In Abschnitt 2.5.4 wurde bereits die Vorgangsweise für das Durchführen einer Simulationsreihe erläutert. Die dafür notwendigen Einstellungen können über ein eigenes Fenster, dargestellt in Abbildung 2.10, vorgenommen werden.



Abbildung 2.10: Fenster mit den Einstellungen für die Durchführung von Simulationsreihen

Simulationsmodus

Legt fest, ob eine Einzelsimulation oder eine Simulationsreihe durchgeführt werden soll. Für eine Simulationsreihe muss eine Job-Liste definiert werden, die die einzelnen Parameterdateien und eventuell Frequenzdateien für die einzelnen Simulationsschritte festlegt (siehe Abschnitt 2.5.4)

Simulationseinstellungen aus Parameterdatei laden

Unabhängig vom Simulationsmodus. Ist diese Option aktiv, werden auch die Simulationseinstellungen aus der Parameterdatei geladen (siehe Abschnitt 2.1.1).

Simulationseinstellungen des Hauptfensters verwenden

Unabhängig vom Simulationsmodus. Ist diese Option aktiv, werden die Simulationseinstellungen des Hauptfensters für alle Simulationsdurchläufe herangezogen.

Wenn möglich, immer gleiche Geräteinstanzen verwenden

Ist diese Option aktiv, werden für jeden Simulationsdurchlauf die gleichen Geräteinstanzen verwendet, sofern dies die Einstellungen ermöglichen.

Wenn möglich, gleiche DSM-Instanzen verwenden

Ist diese Option aktiv, werden für jeden Simulationsdurchlauf die gleichen Geräteinstanzen und DSM_Device-Instanzen verwendet, sofern dies die Einstellungen ermöglichen.

Unterschiedliche Frequenzdaten verwenden

Ist diese Option aktiv, wird neben der Parameterdatei für einen Simulationsdurchlauf auch eine Frequenzdatendatei in der Job-Liste einer Simulationsreihe angegeben. So können verschiedene Frequenzverläufe bei den einzelnen Simulationsdurchläufen einer Simulationsreihe herangezogen werden.

Gleiche Parameterdatei verwenden

Ist diese Option aktiv, werden in der Job-Liste für eine Simulationsreihe nur einmal eine Parameterdatei und verschiedene Frequenzdatendateien festgelegt. Diese Option ist nur dann verfügbar, wenn die Option „Unterschiedliche Frequenzdaten verwenden“ aktiviert ist. So können verschiedene Frequenzverläufe bei gleichen Geräteparametern für die einzelnen Simulationsdurchläufe einer Simulationsreihe herangezogen werden.

Neue Job-Liste

Eine neue Job-Liste erstellen.

Job-Liste laden

Eine vorhandene Job-Liste laden.

Job-Liste speichern

Die aktuelle Job-Liste speichern.

Parameterdatei hinzufügen

Eine Parameterdatei der aktuellen Job-Liste hinzufügen. Ist die Option „Unterschiedliche Frequenzdaten verwenden“ aktiv, werden zuerst eine Parameterdatei und daran anschließend eine Frequenzdatendatei ausgewählt, die zu der aktuellen Job-Liste hinzugefügt werden sollen. Ist Weiters noch die Option „Gleiche Parameterdatei verwenden“ aktiv, muss nur einmal zu Beginn eine Parameterdatei ausgewählt werden.

Job-Liste anzeigen

Wurde die aktuelle Job-Liste gespeichert, kann sie über diese Schaltfläche geöffnet werden, um die Einträge zu kontrollieren bzw. zu bearbeiten.

Beenden

Übernehmen der Einstellungen und Rückkehr zum Hauptfenster.

3. Ermittlung der Parameter für das Simulationsmodell

Für die im Zuge dieser Arbeit durchgeführten Simulationen musste zuerst ein Modell gefunden werden, dass in irgendeiner Weise das Verhalten von Verbrauchern in der Realität möglichst gut nachbilden kann. Im Abschnitt 2.2 wurde darauf eingegangen, wie die individuellen Geräte in ihrer jeweiligen Gerätekategorie modelliert wurden. Mit Hilfe der definierbaren Parameter ist es nun möglich, wie schon im Abschnitt 2.2.6.1 ausgeführt, das Verhalten vieler individueller Verbraucher einem gewünschten Verlauf anzupassen.

Mit Hilfe von Lastprofilen, die das Verhalten einer Masse von Verbrauchern beschreiben, ist es nun möglich, die Parameter für das Simulationsmodell zu ermitteln. Als Grundlage für diese Parameterfindung dient das durchschnittliche Lastprofil, das in [1] ermittelt wurde. In den folgenden Abschnitten soll nun dieses Lastprofil sowie die Ermittlung der Geräteparameter aus diesem erläutert werden.

3.1. Vergleichsdaten

In [1] und [7] wurde versucht, eine Abschätzung des Potentials von DSM in Haushalten anzugeben. Im Rahmen dieser Bemühungen musste auch der Tageslastgang von verschiedenen Verbrauchergruppen ermittelt werden. Mit Hilfe von Statistiken und Datenerhebungen konnte eine Tageslastganglinie angegeben werden, welche in Diagramm 3.1 dargestellt ist.

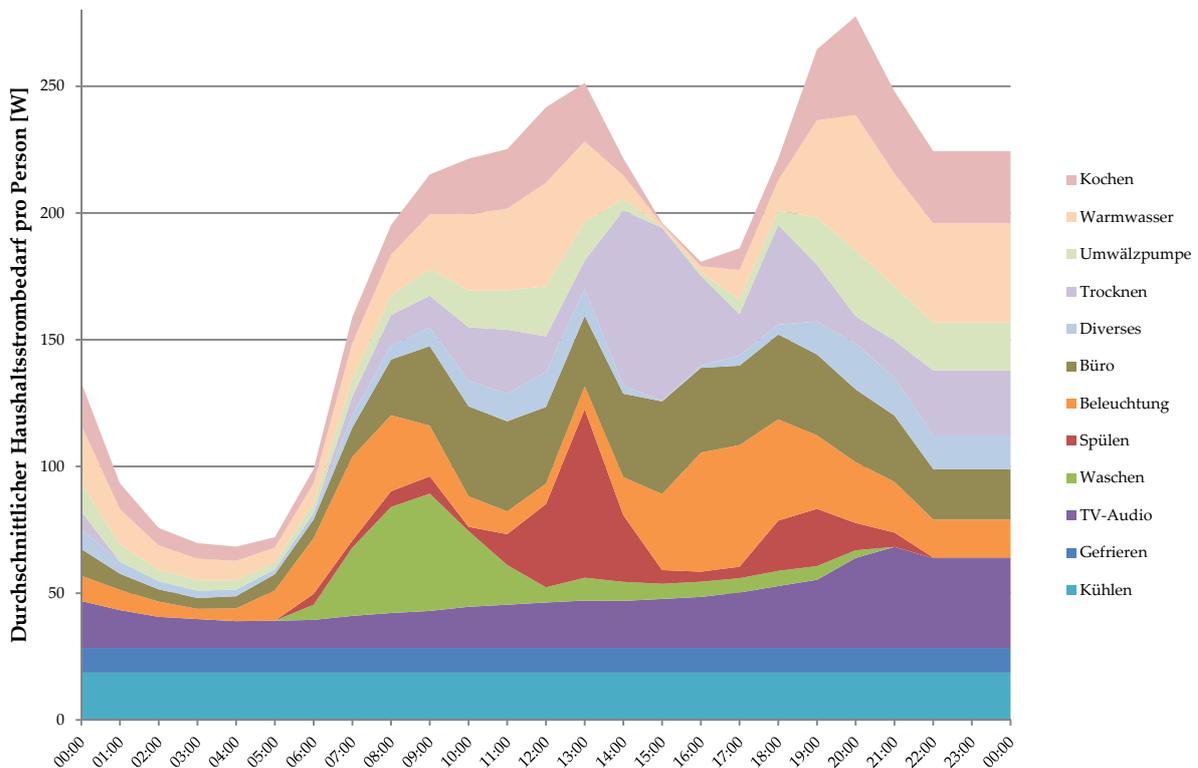


Diagramm 3.1: Summenlastganglinie über die einzelnen Sektoren (nach [1: S.96])

Weiters wurde die Aufteilung des durchschnittlichen Bedarfs an elektrischer Energie eines Haushaltes bzw. einer Person auf diese einzelnen Sektoren ermittelt, die in Tabelle 3.1 angeführt sind.

Für die Sektoren Kochen, Warmwasser, Umwälzpumpe und Diverses konnten keine Tagesverläufe im Stundenraster ermittelt werden. Daher wurden diese Sektoren, gemäß ihrem Anteil am Gesamtverbrauch, auf die Differenz der anderen Sektoren zu der gemittelten Tageslastganglinie eines Haushaltes aufgeteilt [1: S.96].

Die gemittelte Tageslastganglinie ergibt sich aus dem gewichteten Mittelwert der einzelnen Standardlastprofile (H0-Standardlastprofile) eines Haushaltes für die verschiedenen Wochentage und Jahreszeiten.

Im Rahmen dieser Arbeit wurden nicht alle in [1] angeführten Sektoren aufgegriffen. Jene Sektoren, die behandelt werden, sind Kühlen und Gefrieren, TV-Audio, Waschen, Spülen, Beleuchtung und Büro (dunkle Flächen in Diagramm 3.1). Mit diesen Sektoren werden 60% des Bedarfs an elektrischer Energie eines Haushaltes abgedeckt. Eine Einbindung der restlichen Sektoren in die Simulation ist jedoch leicht möglich, da bereits alle notwendigen Klassen für diese Gerätegruppen definiert sind, was aus den Ausführungen in Abschnitt 2.2 hervorgeht.

	Jahresverbrauch			Mittlere Leistung		
	Gesamt	Haushalt	Person	Gesamt	Haushalt	Person
	TWh/a	kWh/a	kWh/a	MW	W	W
Büro	15,8	398	192	1.807	45,4	22,0
Warmwasser	14,9	376	182	1.706	42,9	20,7
TV/Audio	14,5	364	176	1.653	41,6	20,1
Beleuchtung	14,4	361	175	1.640	41,2	19,9
Kühlen	13,4	338	163	1.535	38,6	18,6
Trocknen	13,1	329	159	1.495	37,6	18,2
Kochen	10,9	274	132	1.243	31,3	15,1
Umwälzpumpe	7,2	182	88	826	20,8	10,0
Gefrieren	7,0	177	86	805	20,2	9,8
Spülen	7,0	175	85	796	20,0	9,7
Waschen	6,6	167	80	756	19,0	9,2
Diverses	5,1	128	62	579	14,6	7,0
Summe	130	3.269	1.579	14.840	373,2	180,3

Tabelle 3.1: Bedarf an elektrischer Energie in Deutschland (2006) (aus [1: S.12])

Die nun in weiterer Folge verwendeten Daten aus [1] sind sicherlich mit gewissen Unsicherheiten behaftet. Genauere Daten bezüglich der Aufteilung des elektrischen Energieverbrauchs konnten aber im Zuge der Recherche für diese Arbeit nicht gefunden werden.

Es ergeben sich für die Parameter, die aus diesen Daten gewonnen werden, gewisse Einschränkungen. So werden sich die jahreszeitlichen und von den Wochentagen abhängigen Schwankungen im Energieverbrauch nicht im Geräteverhalten widerspiegeln. Auch die Veränderungen der Gerätezusammenstellung, die seit der Erhebung der verwendeten Statistiken stattgefunden hat, können in der Simulation nicht berücksichtigt werden. Gerade in den Bereichen TV-Audio und Büro kann dahingehend von einer höheren Abweichung ausgegangen werden.

Dennoch erlaubt die Verwendung dieser Daten, eine Modellierung der Geräte für die Simulation zu erhalten, die in gewisser Weise die Realität abbildet. Diese Modellierung dient nun, unabhängig von ihrer Relevanz in Bezug auf reale Geräte, als Grundlage für die verschiedenen Simulationen und erlaubt die Auswirkungen des Einflusses von DSM auf diese spezielle Gerätezusammenstellung zu untersuchen. Von besonderem Interesse sind hier die prinzipiellen Mechanismen, die die Beeinflussung der Leistungsaufnahme begleiten.

Für eine bessere Abbildung der Auswirkungen auf die Realität müssten einerseits neuere und feinere Daten zu den Geräteverteilungen und deren Einsatz verwendet werden. Andererseits müsste auch das Simulationsmodell weiter verfeinert werden, um z.B. unterschiedliche Wochentage oder gar Jahreszeiten berücksichtigen zu können. Dies würde aber den Rahmen dieser Arbeit sprengen.

Mit Hilfe der hier vorgestellten Vergleichsdaten werden in den folgenden Abschnitten die Parameter für die einzelnen Gerätegruppen ermittelt. Dabei wurde versucht, so weit als möglich jene Statistiken und Werte zu verwenden, die auch in [1] herangezogen wurden, um so eine gute Übereinstimmung mit den verwendeten Vergleichsdaten zu erhalten.

3.2. Bürogeräte

Die Bürogeräte werden mit Hilfe der Klasse `Probable_Operation` nachgebildet. Die ersten Parameter, die für diesen Sektor abgeschätzt werden müssen, sind der Stand-by-Verbrauch und die Anschlussleistung. Daran anschließend können, mit Hilfe der Referenzkurve, die Einschaltzeitpunkte, deren Streuung, Startwahrscheinlichkeit und typische Laufzeit ermittelt werden.

3.2.1. Durchschnittlicher Stand-by-Verbrauch

In [1: S.19] wird anhand der verkauften Bürogeräte in Deutschland im Jahr 2007 in Verbindung mit dem bekannten Ausstattungsgrad der Haushalte mit Fernsehern sowie deren Verkaufszahlen auf den Ausstattungsgrad mit verschiedenen Bürogeräten geschlossen.

Weiters wird in [1: S.18] der Stand-by-Verbrauch des Sektors Büro mit 3,2 Watt pro Person angegeben.

Geräte-Typ	Verkaufte Geräte in Tausend	pro Person
TV-Display Total	5.992	0,53
Büro Gesamt	12.707	1,12
Desktop PCs	1.407	0,12
Notebooks	3.010	0,27
Monitore	3.441	0,30
Drucker	4.603	0,41
Scanner	246	0,02

Tabelle 3.2: Verkaufte Bürogeräte in Deutschland 2007 mit Aufschlüsselung auf Geräte pro Person. Aus [1: S.19]

Der Ausstattungsgrad mit Bürogeräten ist 112% für eine Person bzw. 231,8% für einen Haushalt (das heißt jede Person besitzt im Schnitt 1,12 dieser Geräte). Da dieser Wert nahe eins ist, kann einfach der durchschnittliche Stand-by-Verbrauch pro Gerät gleich dem pro Person gesetzt werden. Der Stand-by-Verbrauch tritt nur dann auf, wenn das Gerät nicht aktiv genutzt wird. Man kann aber davon ausgehen, dass immer irgendein Gerät im Einsatz ist, wodurch diese Annahme gerechtfertigt erscheint.

3.2.2. Durchschnittliche Anschlussleistung

Die Ermittlung der durchschnittlichen Anschlussleistung ist in diesem Sektor schwierig, da die einzelnen Gerätetypen sehr unterschiedliche Leistungen aufweisen. Dieser Umstand wird in einer großen Streuung der mittleren Leistung berücksichtigt (siehe Tabelle 3.4).

Da die Bürogeräte als eine einzige Geräteklasse angesehen werden, können nicht mehr als 100% von Ihnen gleichzeitig aktiv sein. Daher ist die durchschnittliche Anschlussleistung etwas höher zu wählen, um auf den gleichen Lastgang zu kommen. Für eine genauere Betrachtung könnte dieser Sektor in die einzelnen Gerätetypen aufgeteilt werden, wovon aber im Rahmen dieser Arbeit abgesehen wurde.

Die Abschätzung der Leistungsaufnahme von Desktop-PCs wurde mit den Daten der Tabelle in [1: S.117] ermittelt. Darin sind der minimale und maximale Stromverbrauch von 16 sparsamen und 8 handelsüblichen Desktop-PCs aufgelistet. Aus den Maximalwerten wurde der Mittelwert gebildet und für die durchschnittliche Leistungsaufnahme dieser Geräte gesetzt.

Für die Gerätegruppe der Notebooks wurde die Tabelle mit Angaben der Leistungsaufnahme von neun 15,4-Zoll-Notbooks in [1: S.25] herangezogen. Aus den Daten bei Netzbetrieb und aktiven Notebook (die Betriebszustände „Suspend“ und „ausgeschaltet“ wurden daher nicht berücksichtigt) wurde die durchschnittliche Leistungsaufnahme berechnet, da diese gut alle möglichen Anwendungszustände eines Notebooks widerspiegeln.

In [1: S.18] ist die Leistungsaufnahme von verschiedenen Peripheriegeräten aufgelistet. Aus den Maximalwerten wurde der Mittelwert für den jeweiligen Gerätetyp ermittelt. In [16] und [17] wird der Anteil von Tintenstrahldruckern an den Verkaufszahlen mit ca. 66% angegeben. Der Rest sind Laserdrucker. Wenn man davon ausgeht, dass Laserdrucker eine etwas höhere durchschnittliche Lebensdauer aufweisen als Tintenstrahldrucker (Annahme hierzu 15% längere Lebensdauer) ergibt sich, dass bei den aktiven Geräten ca. 57% der Drucker Tintenstrahl-Geräte sind. Dementsprechend werden die Anteile auf diese beiden Gerätegruppen aufgeteilt.

Für sonstige Geräte (Scanner, Router, Modem, Beamer, ...) wurde ein Wert von 80 W angenommen. Durch den geringen Anteil (0,02 Geräte pro Person) nimmt diese Leistungsangabe trotz der großen Unsicherheit ihres Wertes kaum Einfluss auf das Gesamtergebnis.

Tabelle 3.3 fasst die ermittelten Werte zusammen. In Verbindung mit den in Tabelle 3.2 ermittelten Gerätzahlen pro Person kann die Anschlussleistung der einzelnen Gerätegruppen pro Person ermittelt werden. Die Summe dieser Werte ergibt die Anschlussleistung pro Kopf für den gesamten Sektor Büro.

Gerätetyp	Geräte pro Person	Ø-Leistungsaufnahme pro Gerät	Ø-Anschlussleistung pro Person
Desktop PCs	0,12	79,38 W	9,53 W
Notebooks	0,27	40,52 W	10,94 W
Monitore	0,30	63,29 W	18,99 W
Drucker (Tinte)	0,23	18,00 W	4,21 W
Drucker (Laser)	0,18	707,33 W	124,70 W
Sonstiges	0,02	80,00 W	1,60 W
Summe:			169,96 W

Tabelle 3.3: Ermittlung der durchschnittlichen Anschlussleistung pro Person für Bürogeräte

3.2.3. Zusammenfassung der Parameterwerte

Mit den Werten für Stand-by-Verbrauch und die Anschlussleistung der Bürogeräte wurden die zeitlichen Parameter so gewählt, dass sich die Simulationsergebnisse von Bürogeräten von vielen Personen mit der Vergleichskurve übereinstimmen.

Power_Nominal	170	W	30	%			
Power_Stand_by	3,2	W	30	%			
Time_Start	05:00	09:30	13:45	14:30	17:30	18:00	Uhr
'-> Std.Dev.:	120	120	90	50	70	190	min
Start_Probability	2,75	65,5	20	12	6,8	90	%
Time_typ_Run	60	90	90	48	30	90	min
'-> Std.Dev.:	20	%					

Tabelle 3.4: Parameterwerte für Bürogeräte (Auszug aus Parameterdatei)

Tabelle 3.4 stellt die gewählten Parameter dar. Wie schon in Abschnitt 3.2.2 erwähnt, wurde die Streuung von Anschluss- und Stand-by-Leistung mit einem hohem Wert angenommen, da auch die, diese Geräteklasse bildenden Einzelgeräte stark unterschiedliche Leistungsaufnahmen haben.

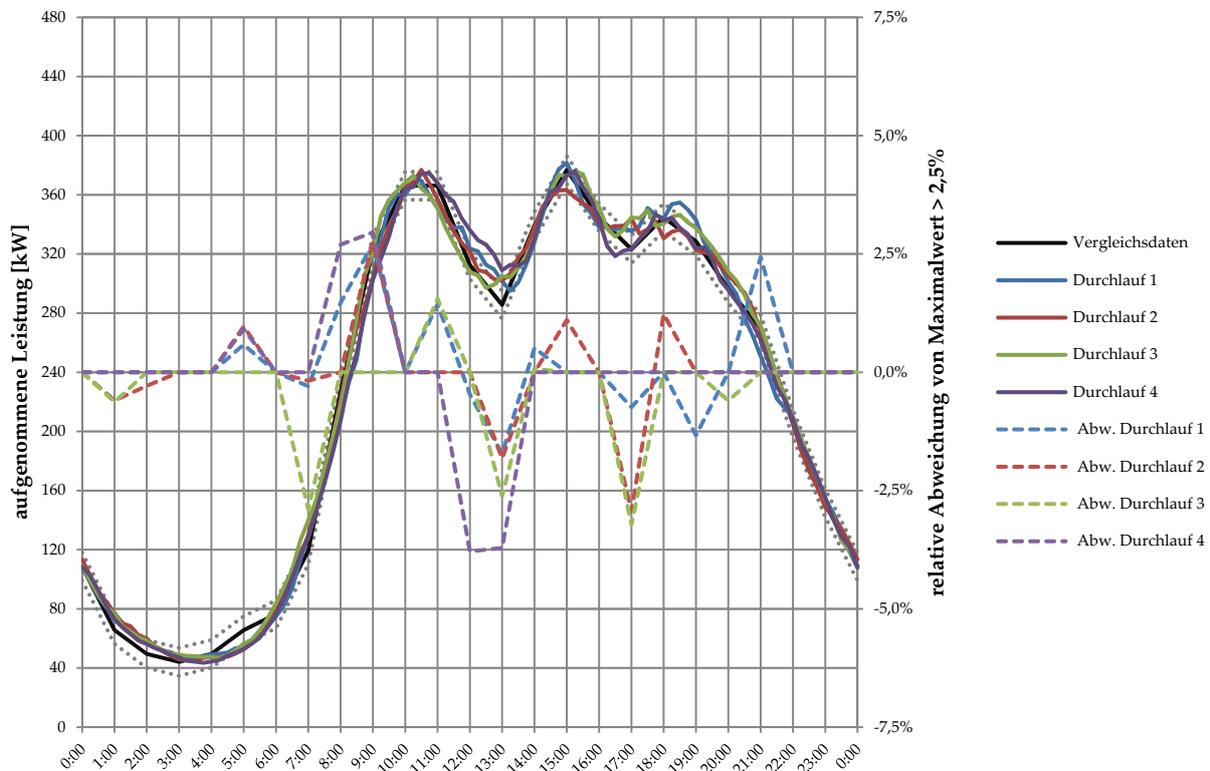


Diagramm 3.2: Ergebnis von vier Simulationdurchläufen mit ca. 10.000 aktiven Bürogeräten mit Abweichungen von den Vergleichsdaten

Mit diesem Parametersatz wurden vier Simulationdurchläufe mit einem Simulationszeitraum von 24h, einer Viertel-Stunden-Auflösung und Geräten von 10.300 Personen durchgeführt. In Diagramm 3.2 sind

die Ergebnisse dieser Durchläufe zusammengefasst. Bei jedem Durchlauf wurden zu Beginn neue Geräteinstanzen erzeugt.

Um eine Bewertung der Übereinstimmung der Simulations- mit den Vergleichsdaten zu erhalten, werden im Folgenden die relativen Abweichungen der Simulation vom Maximalwert der Vergleichsdaten ermittelt und betrachtet. Die Vergleichsdaten liegen als Stundenwerte vor (schwarze Kurve), weshalb die Abweichungen nur zu jeder vollen Stunde ermittelt werden (vertikale Gitternetzlinien). Eine Abweichung von weniger als 2,5% vom Maximalwert der Vergleichsdaten wird nicht berücksichtigt, da dies als ausreichende Übereinstimmung angesehen wird. Diesen Bereich kennzeichnen die beiden grauen Kurven ober- und unterhalb der Vergleichsdaten. Jene Abweichungen von den Vergleichsdaten, die größer als 2,5% sind, sind strichliert eingetragen.

Die Durchführung mehrerer Simulationsdurchläufe mit verschiedenen Geräteinstanzen soll helfen, jene Abweichungen von den Vergleichsdaten, die aufgrund eines ungenauen Modells bzw. der unzureichend gewählten Parameter auftreten, von jenen zu unterscheiden, die zufällig aufgrund der geringen Anzahl an Geräten entstehen. Erstere sind daran erkennbar, dass zu den gleichen Zeitpunkten immer eine Abweichung derselben Größenordnung auftritt, oder anders gesagt die Abweichungen zueinander stark korreliert sind. Hier müsste das Modell bzw. die Parameter angepasst werden. Mit vier Simulationsdurchläufen lässt sich dahingehend zumindest eine grobe Abschätzung angeben.

Gut zu erkennen ist, dass durch die Parametervariierung immer andere Sets von Verbrauchern erzeugt werden, die in Summe ein dem gewünschten Verlauf ähnliches Verhalten aufweisen. Größere Abweichungen gibt es nur bei den steilen Teilen der Kurvenverläufe (z.B. von 8:00 bis 9:00 Uhr), sowie der verringerten Last um 13:00, die nicht komplett nachgebildet werden können. Auch um 5:00 ist eine kleine Abweichung von den Vergleichsdaten feststellbar, die aber deutlich geringer ausfällt.

Die restlichen Abweichungen korrelieren nicht miteinander sondern sind nur zufällige Abweichungen der erzeugten Geräteinstanzen. Je mehr Geräte simuliert werden, desto mehr gleichen sich diese dem Sollverlauf an. Die größten Abweichungen vom Sollverlauf sind im Bereich von 5% vom Maximalwert und sind daher noch immer ausreichend nahe dem gewünschten Ergebnis.

	Durchlauf 1	Durchlauf 2	Durchlauf 3	Durchlauf 4
aktive Geräte	10.077	10.091	10.076	10.053
Ausstattungsgrad	97,83 %	97,97 %	97,83 %	97,6 %
Durchschnittsverbrauch	21,92 W	21,94 W	22,17 W	21,88 W
Abweichung	- 0,38 %	- 0,25 %	+ 0,78 %	- 0,54 %

Tabelle 3.5: Zusammenfassung von ermittelten Werten aus den Simulationsdurchläufen mit Bürogeräten von 10.300 Personen

Tabelle 3.5 fasst weitere, bei den Simulationen ermittelte Daten zusammen. Der Ausstattungsgrad berechnet sich aus den aktiven Geräten pro Person. Dieser bewegt sich nahe 100%, das heißt, dass fast alle Geräte im Tagesverlauf eingeschaltet werden und in der restlichen Zeit im Stand-by-Modus sind. Dieser Wert war aufgrund der eingangs angegebenen Anzahl von Geräten pro Person zu erwarten.

Weiters ist der Durchschnittsverbrauch über die simulierten 24h nahe an den für diese Gerätegruppe in Tabelle 3.1 angegebenen Wert von 22W pro Person. Die relativen Abweichungen der einzelnen Durchläufe von diesem Wert sind in der letzten Zeile angeführt.

3.3. Audio-Video-Geräte

In der Gerätegruppe Audio-Videogeräte wird neben Fernsehern der gesamte weitere Audio-Video-Bereich inkl. Radiogeräte betrachtet. Zur Ermittlung der Parameterwerte wird ähnlich vorgegangen wie bei den Bürogeräten, da auch hier die Geräte Klasse `Probable_Operation` Verwendung findet.

3.3.1. Durchschnittlicher Stand-by-Verbrauch

In [1: S.32] wird, ähnlich wie bei den Bürogeräten, aus den Verkaufszahlen auf die Geräte pro Person geschlossen. Diese Daten wurden dann mit dem typischen Stand-by-Verbrauch der verschiedenen Gerätegruppen aus einer Studie kombiniert, um den Stand-by-Verbrauch des gesamten Sektors zu ermitteln.

Der so in Tabelle 3.6 ermittelte Wert ist deutlich höher als der in [1] dann weiterverwendete Wert von 8,7W, da dieser nicht berücksichtigt, dass Geräte nicht 24h im Stand-by-Betrieb sind. Für die Parameterwahl ist aber genau dieser Wert von Interesse (siehe Abschnitt 2.2.4.1) und wird für diese Gerätegruppe übernommen.

Gerätetyp	Geräte pro Person	Stand-by in W	Stand-by pro Person in W	Stand-by pro Person in Wh/d
TV-Display Total	0,53	7,3	3,9	93
Set Top Boxen	0,38	9,7	3,7	88
Video Total	0,41	10,0	4,1	97
Videogames Konsolen	0,35	1,7	0,6	14
Hi-Fi	0,22	7,0	1,6	38
Radio	0,49	1,4	0,7	16
Gesamt	2,38	Summen:	14,4	346

Tabelle 3.6: Stand-by-Verbrauch im Sektor TV-Audio, aus [1: S.38]

3.3.2. Durchschnittliche Anschlussleistung

Zur Ermittlung der durchschnittlichen Anschlussleistung der Audio-Video-Geräte wird genauso vorgegangen wie bei den Bürogeräten.

In [1: S.35] ist die Leistungsaufnahme von verschiedener Unterhaltungselektronik aufgelistet. Diese Messwerte werden, nachdem der Mittelwert für jede Gerätegruppe ermittelt wurde, mit der Anzahl der Geräte pro Person verknüpft, um so auf die durchschnittliche Anschlussleistung zu kommen, die sich aus der Summe der Einzelgerätegruppen zusammensetzt.

Gerätetyp	Geräte pro Person	Leistung	Anteil
TV-Display Total	0,53	68,33 W	36,22 W
Set Top Boxen	0,38	20,00 W	7,60 W
Video Total	0,41	56,50 W	23,17 W
Videogames Konsolen	0,35	66,00 W	23,10 W
Hi-Fi	0,22	21,90 W	4,82 W
Radio	0,49	1,40 W	0,69 W
Gesamt	2,38	Summe:	95,59 W

Tabelle 3.7: Ermittlung der Anschlussleistung pro Person für Audio-Video-Geräte

Dieser Wert ist nahe dem in [1: S.47] ermittelten Werten für den Sektor TV-Audio (92 W für TV mit Zusatzgeräten, 10 W für Audiogeräte und Gesamt-Stand-by-Leistung von 8,7 W), die auf einem anderem Weg ermittelt wurden.

3.3.3. Zusammenfassung der Parameterwerte

Nach dem Festlegen der Stand-by-Leistung und der Anschlussleistung werden die Zeit-Parameter und Einschaltwahrscheinlichkeiten so gelegt, dass es wieder zu einer Übereinstimmung mit den Vergleichsdaten kommt.

Power_Nominal	96	W	20	%					
Power_Stand_by	14	W	20	%					
Time_Start	09:39	12:09	16:56	18:26	20:03	21:32	22:09	Uhr	
'-> Std.Dev.:	180	180	120	60	55	30	60	min	
Start_Probability	10	20	36	12	42	5	10,5	%	
Time_typ_Run	141	141	94	94	122	78	141	min	
'-> Std.Dev.:	20	%							

Tabelle 3.8: Parameterwerte für Audio-Video-Geräte (Auszug aus Parameterdatei)

Mit den Parametern, die in Tabelle 3.8 zusammengefasst sind, wurden wieder vier Simulationsdurchläufe über einen Nutzungszeitraum von 24h mit Audio-Video-Geräten von 13.600 Personen durchgeführt. Die Ergebnisse dieser Durchläufe sind in Diagramm 3.3 mit den Abweichungen vom Maximalwert zu den Vergleichsdaten zusammengefasst.

Mit den gewählten Parametern konnte eine sehr gute Übereinstimmung mit den Vergleichsdaten erzielt werden. Nur im Bereich von 4:00 bis 6:00 kommt es zu einer größeren Abweichung, die aufgrund eines

etwas zu hoch gewählten Stand-by-Verbrauchs zustande kommt. Da diese Abweichung aber geringer als 2,5% vom Maximalwert dieser Gerätekategorie ist, wird der Verlauf als ausreichend genau betrachtet.

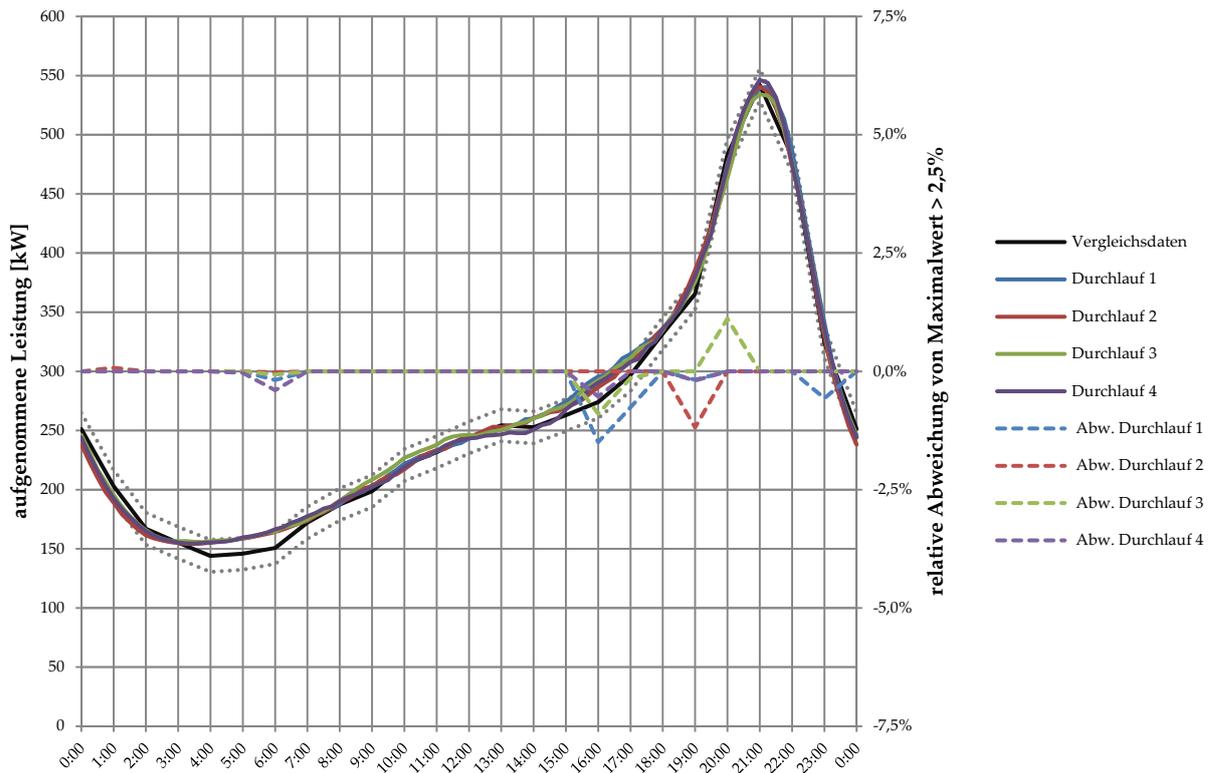


Diagramm 3.3: Ergebnis von vier Simulationsdurchläufen mit ca. 10.000 aktiven Audio-Video-Geräten mit Abweichungen von den Vergleichsdaten

Ein interessantes Detail der gewählten Parameter ist der Umstand, dass sich die Summe aus den mit der Startwahrscheinlichkeit gewichteten typischen Laufzeiten aller Startzeitpunkte zu 157,4 min ergibt. Diese kann als Nutzungszeit T_{Nutzung} für die Audio-Video-Geräte angesehen werden, da es der Zeit entspricht, die jedes Gerät im Durchschnitt am Tag in Betrieb ist:

$$T_{\text{Nutzung}} = \sum_{i=1}^n \left(\frac{P_{\text{Time_Start}}(i)}{\text{Start_Probability}} \cdot T_{\text{Time_typ_Run}}(i) \right) \quad (3.1)$$

Der Verlauf der Vergleichsdaten kommt aus der Nutzung von TV-Geräten zustande ([1: S.46]). Für diese wird dort eine durchschnittliche Nutzungszeit von 157 min pro Person angegeben. Das ist eine bemerkenswert gute Übereinstimmung der gewählten zeitlichen Parameter und Startwahrscheinlichkeiten mit der Statistik, auf der die Vergleichsdaten beruhen.

	Durchlauf 1	Durchlauf 2	Durchlauf 3	Durchlauf 4
aktive Geräte	10.839	10.817	10.886	10878
Ausstattungsgrad	79,70 %	79,54 %	80,07 %	79,99 %
Durchschnittsverbrauch	19,74 W	19,60 W	19,72 W	19,62 W
Abweichung	1,60 %	0,86 %	1,51 %	0,99 %

Tabelle 3.9: Zusammenfassung von ermittelten Werten aus den Simulationsdurchläufen mit Audio-Video-Geräten von 13.600 Personen

Tabelle 3.9 gibt einen Überblick über weitere, während der Simulation ermittelte Daten. Der Ausstattungsgrad beträgt ca. 80%. Dies ist auf den ersten Blick überraschend, da in Tabelle 3.6 eine Gesamt-Geräteanzahl von 2,38 pro Person ermittelt wurde. Nur wird hier angenommen, dass diese Geräte meist gemeinsam von mehreren Personen gleichzeitig genutzt werden, wodurch die Anschlussleistung in Tabelle 3.7 zustande kommt.

Der Durchschnittsverbrauch dieser Gerätegruppe liegt nach Tabelle 3.1 bei 19,4 W. Dieser Wert wird bei dem gewählten Parametersatz leicht überschritten, befindet sich aber in einem ausreichend genauem Bereich.

3.4. Beleuchtung

Bei der Beleuchtung gibt es keine klaren Daten bezüglich Ausstattungsgrad, der aber sicherlich höher als 100% ist, da fast jeder Haushalt mehr als eine Lampe besitzt.

Für die Simulation wurden daher Verbraucher mit einer durchschnittlichen Leistung von 200W herangezogen, weil mit diesem Wert alle „Geräte“ aktiv werden und den Verlauf der Vergleichsdaten nachfolgen können. Dies führt dazu, dass bei Betrachtung weniger Personen in diesem Sektor bezüglich der Verteilung der Aktivitätszeiten eine hohe Unsicherheit besteht.

Bezüglich des Beitrages dieser Gerätegruppe zu einem reduzierten Lastgang wird in [1: S.56] nur von einer Reduzierung der Helligkeit ausgegangen. Für diesen Fall ist es nicht wichtig, wie viele Einzellampen das bewerkstelligen, sondern die Summe aus diesen. Falls aber in kritischen Netzsituationen bestimmte Beleuchtungselemente abgeschaltet werden, um sozusagen mit den verbleibenden Lampen eine Art Notbeleuchtung aufrecht zu erhalten, können in Simulationen mit wenigen Personen keine feingestrukturierten Daten ermittelt werden.

Für eine bessere Auflösung dieses Sektors könnte diese Gerätegruppe in mehrere Sub-Gerätegruppen aufgeteilt werden (z.B. in Glühlampen, Leuchtstofflampen, Kompaktleuchtstofflampen, sonstige Beleuchtung). Die Aufteilung könnte hier nach den aktuellen Verkaufszahlen in Kombination mit der typischen Lebenserwartung einer Lampe ermittelt werden. Im Rahmen dieser Arbeit wird aber davon abgesehen.

Power_Nominal	200	W	20	%			
Time_Start	23:45	07:00	11:00	11:00	16:00	19:15	Uhr
'-> Std.Dev.:	120	90	45	240	90	120	min
Start_Probability	13	81	3	17	86	46	%
Time_typ_Run	85	42	30	120	55	70	min
'-> Std.Dev.:	20	%					

Tabelle 3.10: Parameterwerte für die Beleuchtung (Auszug aus Parameterdatei)

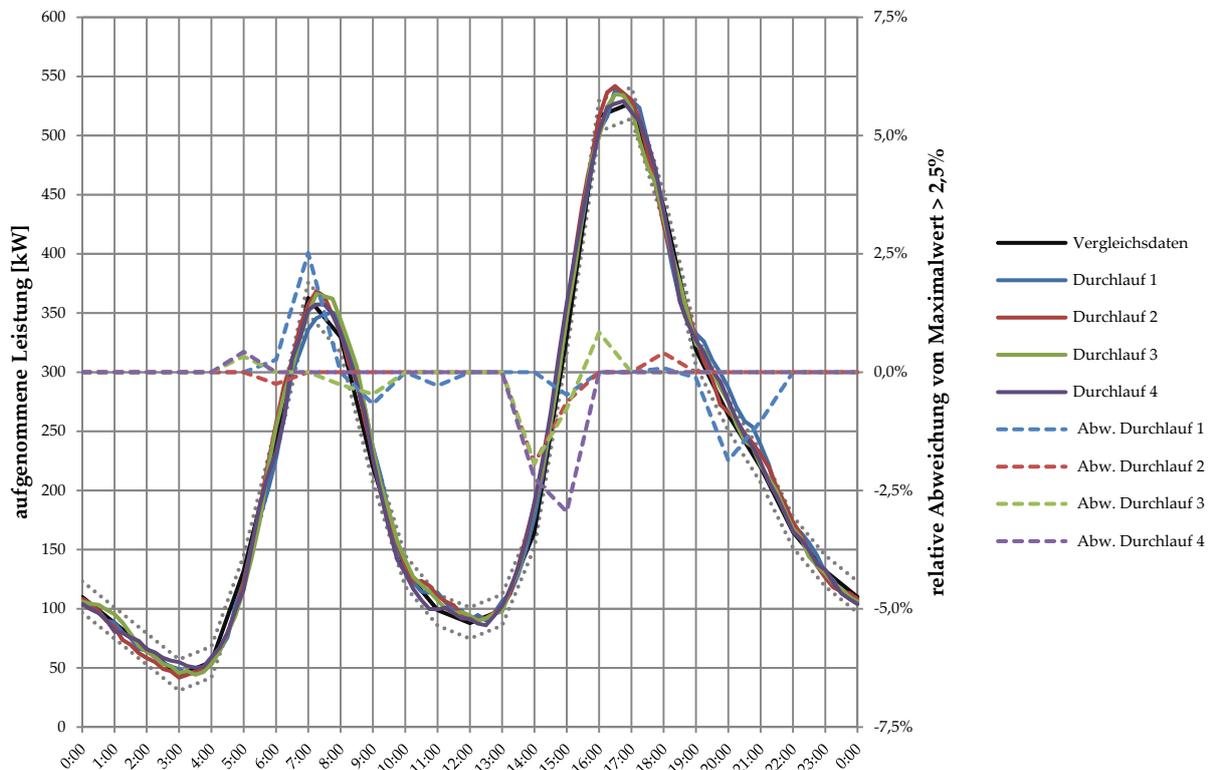


Diagramm 3.4: Ergebnis von vier Simulationsdurchläufen mit ca. 10.000 aktiven Beleuchtungskörpern mit Abweichungen von den Vergleichsdaten

Tabelle 3.10 fasst die gewählten Parameterwerte zusammen. Ähnlich wie bei den Audio-Videogeräten kann mit den ermittelten Parameterwerten nach Gleichung (3.1) eine durchschnittliche Nutzungszeit T_{Nutzung} für Beleuchtung mit 145,9 min berechnet werden. Dieser Wert liegt in der Größenordnung der Nutzungszeit von Audio-Video-Geräten (154 min). Zwar lassen sich diese Daten nicht wirklich vergleichen, weil sie aus unterschiedlichen Statistiken aus unterschiedlichen Jahren kommen, es liegt aber der Schluss nahe, dass sich die beiden Nutzungszeiten ähneln sollten, wenn man davon ausgeht, dass z.B. beim Fernsehen auch die Beleuchtung eingeschaltet wird und der Bereich Audio-Video einen Großteil der Freizeitnutzung in der privaten Wohnung bzw. im eigenen Haus nachbildet. Leider liegen gerade aus diesem Sektor keine genaueren Statistiken vor, um die getroffenen Annahmen verifizieren zu können.

Mit den Parameterwerten aus Tabelle 3.10 wurden wiederum vier Simulationsdurchläufe gemacht. Dabei wurde der Verbrauch für Beleuchtung von 11.000 Personen simuliert.

In Diagramm 3.4 sind die Ergebnisse dieser Durchläufe sowie deren Abweichungen von den Vergleichsdaten zusammengefasst. Es ist zu erkennen, dass mit den gewählten Parametern die Vergleichsdaten relativ gut nachgebildet werden können. Die auftretenden Abweichungen sind nicht miteinander korreliert und entstehen durch zufällige Abweichungen der erzeugten Geräteinstanzen. Werden mehr Einzelgeräte simuliert, nehmen auch diese Abweichungen immer mehr ab.

	Durchlauf 1	Durchlauf 2	Durchlauf 3	Durchlauf 4
aktive Geräte	10.895	10.893	10.883	10.904
Ausstattungsgrad	99,05 %	99,03 %	98,94 %	99,13 %
Durchschnittsverbrauch	19,64 W	19,69 W	19,65 W	19,55
Abweichung	0,81 %	1,05 %	0,86 %	0,35 %

Tabelle 3.11: Zusammenfassung von ermittelten Werten aus den Simulationsdurchläufen mit der Beleuchtung von 11.000 Personen

Die Ausstattungsgrade in Tabelle 3.11 zeigen, dass nahezu alle Beleuchtungskörper aktiv werden, weshalb die durchschnittliche Anschlussleistung pro Person im Schnitt den angenommen 200W entspricht. Ein geringerer Wert hätte dazu geführt, dass die Maxima in Diagramm 3.4 nicht erreicht werden.

Auch der Durchschnittsverbrauch ist nahe dem in Tabelle 3.1 angegebenen Wert von 19,5 W pro Person.

3.5. Kühlgeräte

Die im Simulationsmodell verwendeten Kühlgeräte werden durch die Klasse `Thermal_Storage` repräsentiert. Neben der Anschlussleistung werden die charakteristischen thermischen Parameter (thermischer Widerstand und Kapazität) benötigt. Diese Daten sind aber kaum für diese Geräte angegeben bzw. konnten nicht in passender Form recherchiert werden. Es müssen daher diese Parameter aus den vorhandenen Daten ermittelt werden.

3.5.1. Ermittlung der thermischen Parameter

Die meisten Daten, die im Zuge der Recherche über Kühlgeräte gefunden werden konnten, bezogen sich auf deren relative Einschaltdauer α in Bezug auf die Periodendauer T eines typischen Kühlvorgangs [3, 18]:

$$T_{On} = \alpha \cdot T \quad (3.2)$$

Weiters kann die Anschlussleistung eines Kühlgeräts P_{el} als bekannt vorausgesetzt werden. Bei Kühlgeräten wird thermische Energie dem Speicher entzogen, was hier durch ein negatives Vorzeichen berücksichtigt wird:

$$P_{On} = -\eta \cdot P_{el} \quad (3.3)$$

Um aus diesen Daten Werte für den thermischen Widerstand R_{th} und die thermische Kapazität C_{th} zu ermitteln, wird auf das einfache Modell eines thermischen Speichers zurückgegriffen (siehe Abschnitt 2.2.3). Dort wurde die Änderung der Temperatur im Speicher ermittelt (Gleichung (2.8)):

$$\frac{d\vartheta(t)}{dt} = \frac{1}{R_{th}C_{th}} (P_Z(\vartheta) \cdot R_{th} + \vartheta_A - \vartheta(t)) \quad (3.4)$$

Für eine einfachere Berechnung bietet sich eine Linearisierung der Temperaturänderung um die Solltemperatur ϑ_{Set} an. Es werden nun nur noch die (linearen) Änderungen der Temperatur betrachtet. Dabei sind die beiden auftretenden Betriebszustände zu unterscheiden:

1. Kühlaggregat aktiv

Wenn das Kühlaggregat aktiv wird, hat die Temperatur die obere Schranke erreicht. Nun wird um ϑ_{SP} abgekühlt, das heißt solange, bis die untere Schaltschwelle erreicht wird. Dies soll in der üblichen Laufzeit T_{On} erfolgen. Somit ergibt sich aus (3.4):

$$\left. \frac{d\vartheta(t)}{dt} \right|_{P_Z = -\eta P_{el}} = \frac{\vartheta_{SP}}{T_{On}} = \frac{\vartheta_{SP}}{\alpha T} = \frac{1}{R_{th} C_{th}} (\vartheta_A - \vartheta_{Set} - \eta P_{el} R_{th}) \quad (3.5)$$

2. Kühlaggregat inaktiv

Bei inaktivem Kühlaggregat erwärmt sich der thermische Speicher (da $\vartheta_A > \vartheta_{Set}$) langsam. Bis er wieder aktiv wird, das heißt die obere Schaltschwelle erreicht wird, vergeht die restliche Zeit der Periodendauer:

$$\left. \frac{d\vartheta(t)}{dt} \right|_{P_Z = 0} = \frac{\vartheta_{SP}}{T_{Off}} = \frac{\vartheta_{SP}}{(1-\alpha)T} = \frac{1}{R_{th} C_{th}} (\vartheta_A - \vartheta_{Set}) \quad (3.6)$$

Aus (3.5) und (3.6) lassen sich nun Ausdrücke für den thermischen Widerstand und die thermische Kapazität ermitteln:

$$R_{th} = \frac{(\vartheta_A - \vartheta_{Set})}{\eta \alpha P_{el}} \quad (3.7)$$

$$C_{th} = \eta P_{el} \frac{\alpha (1-\alpha) T}{\vartheta_{SP}} \quad (3.8)$$

Diese Näherungswerte sind nur dann gültig, wenn der Temperaturverlauf zwischen den Schaltschwellen annähernd linear ist. Dies impliziert entweder eine kleine Bandbreite der Temperatur um den Sollwert ($\vartheta_{SP} \ll$) oder eine sich von der Umgebungstemperatur deutlich unterscheidende Solltemperatur ($|\vartheta_A - \vartheta_{Set}| \gg 0$). Für die zu modellierenden Kühlgeräte treffen diese Forderungen weitgehend zu.

3.5.2. Kühlschränke

Für Kühlschränke wird eine relative Einschaltdauer α von 0,3 angegeben [3: S.11, 18: S.24].

Pro Person ergibt sich gemäß Tabelle 3.1 eine mittlere Leistung von 18,6 W. Bei einem Ausstattungsgrad von 0,4 Geräten pro Person (im Jahr 2004) ist die mittlere Leistung eines Kühlschranks [1: S.60]:

$$\bar{P}_{\text{Kühlschrank}} = \frac{18,6 \text{ W}}{0,4} = 46,2 \text{ W} \quad (3.9)$$

Daraus lässt sich die durchschnittliche Anschlussleistung eines Kühlschranks angeben:

$$P_{\text{Kühlschrank}} = \frac{\bar{P}_{\text{Kühlschrank}}}{\alpha} = \frac{46,2 \text{ W}}{0,3} = 154 \text{ W} \quad (3.10)$$

Bei Messungen an einem Kühlschrank (siehe Diagramm 3.5) wurden innerhalb einer neuntägigen Messung folgende Parameterwerte ermittelt:

$$P_{\text{Kühlschrank}} = 91,3 \text{ W}$$

$$\bar{P}_{\text{Kühlschrank}} = 34,4 \text{ W}$$

$$\alpha = 0,377$$

$$T = 34,7 \text{ min}$$

In [3: S.11] wird die Anschlussleistung eines Kühlschranks mit 114 W angegeben.

Diese unterschiedlichen Angaben zeigen, dass bei Kühlschränken eine hohe Bandbreite der Parameter vorliegt, was alleine schon ein Blick in einen Fachmarkt erahnen lässt. Es haben auch viele Faktoren Einfluss auf die Leistungsaufnahme (Öffnungshäufigkeit, Umgebungstemperatur, ...). Um dennoch diese Geräteklasse simulieren zu können, wird ein Wert der Anschlussleistung von 154 W, die sich aus den Daten von [1: S.60] in Verbindung mit (3.10) ergibt, sowie eine relative Einschaltdauer von 0,3 gemäß [3: S.11] gewählt, um eine Deckung mit den Vergleichsdaten zu erreichen. Aufgrund der durchgeführten Messung wird für die Simulation eine Periodendauer von 35 min angenommen.

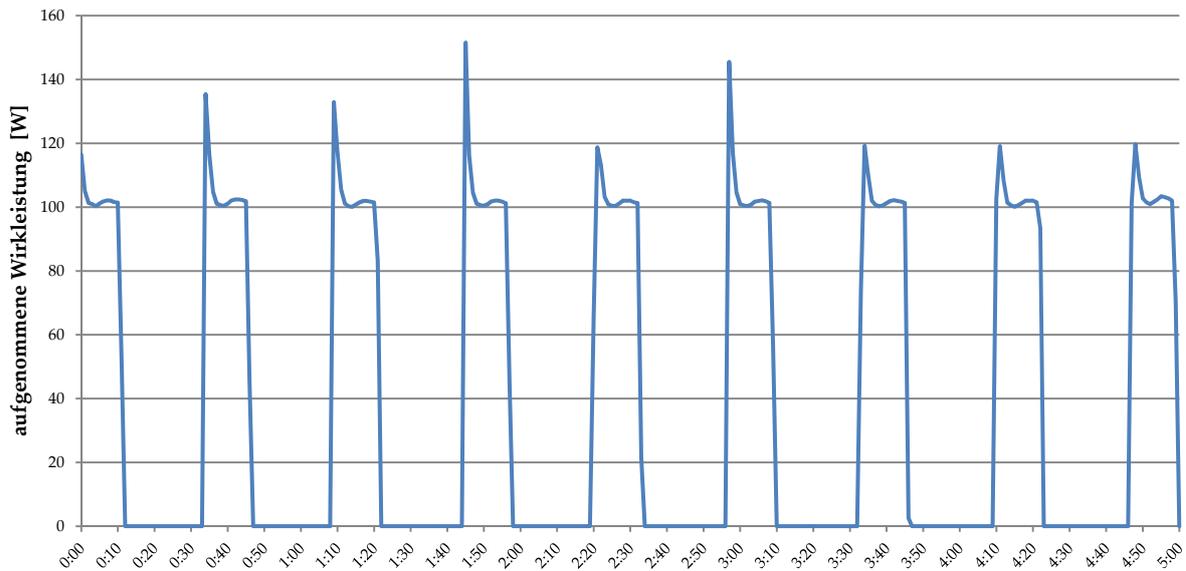


Diagramm 3.5: Wirkleistungsaufnahme eines Kühlschranks „Whirlpool ART 455/4“
Ausschnitt aus der Messung (5h)

Die Solltemperatur in den Kühlschränken wird im Mittel mit 5°C angenommen [1], die Umgebungstemperatur mit 25°C , die Breite um die Solltemperatur mit 4°C (aus Messung in [9: S.50]) und der Wirkungsgrad mit 80% [9]. Mit diesen Daten ergibt sich für die thermischen Parameter des Durchschnittskühlschranks (einsetzen in (3.7) und (3.8)):

$$R_{th} = \frac{1}{0,8 \cdot 154 \text{ W}} \cdot (25^{\circ}\text{C} - 5^{\circ}\text{C}) \cdot \frac{1}{0,3} = 0,541 \frac{\text{K}}{\text{W}} \quad (3.11)$$

$$C_{th} = 0,8 \cdot 154 \text{ W} \cdot \frac{0,3 \cdot (1-0,3) \cdot 35 \text{ min} \cdot 60^{\text{s}}/\text{min}}{4 \text{ K}} = 13.582,8 \frac{\text{J}}{\text{K}} \quad (3.12)$$

Mit diesen Werten wurde nun eine Simulation dieses Kühlschranks durchgeführt, der „den“ Durchschnittskühlschrank repräsentiert. Zusätzlich zur aufgenommenen Leistung wurde auch noch die Temperatur im Kühlschrank ermittelt. Im ersten Versuch wurde das thermische Modell in Sekunden-Schritten simuliert, das Ergebnis ist in Diagramm 3.6 dargestellt (rote Kurven).

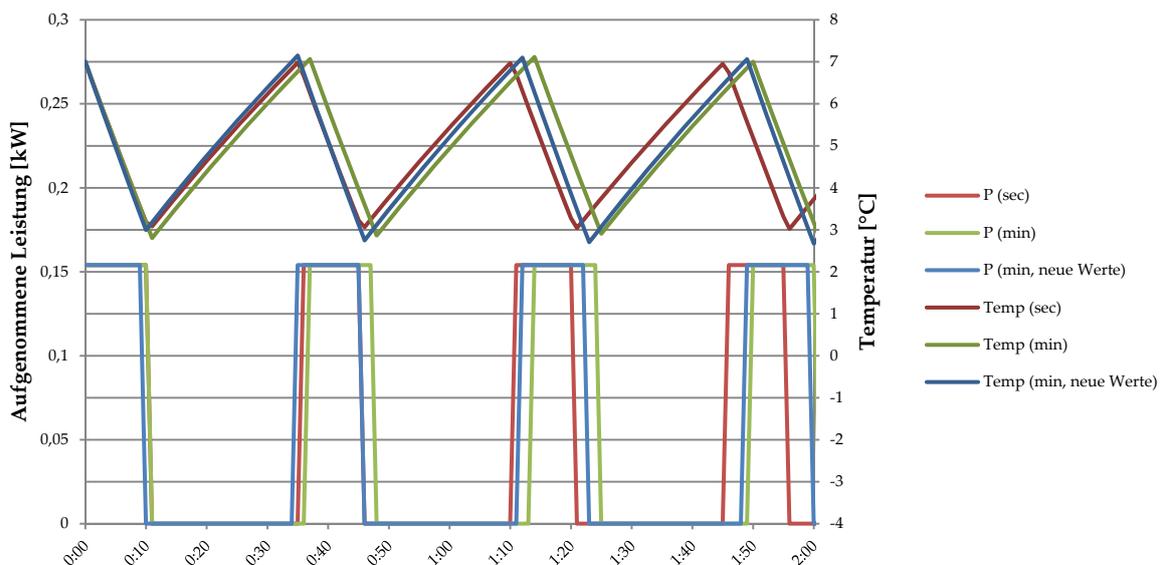


Diagramm 3.6: Vergleich der Auswirkungen der verschiedenen Parameterwerte und Berechnungsmethoden im thermischen Modell des Durchschnittskühlschranks; untere Kurven: Aufgenommene Leistung [kW]; obere Kurven: Temperatur im Kühlschrank [$^{\circ}\text{C}$]

Dann wurde das gleiche thermische Modell in Minuten-Teilung simuliert (grüne Kurven in Diagramm 3.6), was zu einen leicht abgeänderten Ergebnis führt (siehe Abschnitt 2.2.3).

Tabelle 3.12 gibt eine Zusammenfassung der nach der Simulation ermittelten Daten für einen Simulationszeitraum von 24h. Bei einer feinen Auflösung liefert die Näherung des thermischen Modells eine gute Übereinstimmung mit den geforderten Werten. Dies kann man anhand der Temperaturverläufe in Diagramm 3.6 verifizieren, welche im normalen Betrieb relativ linear sind, da die Solltemperatur viel kleiner ist als die Umgebungstemperatur.

	Einheit	Sollwert	sec-Auflösung	Abweichung von Sollwert	min-Auflösung	Abweichung von Sollwert	min-Auflösung (neue Werte)	Abweichung von Sollwert
R_{th}	K/W		0,541		0,541		0,551	
C_{th}	J/K		13.582,8		13.582,8		13.066,4	
$\bar{P}_{\text{Kühlschrank}}$	W	46,2	46,06	- 0,30 %	47,0	+ 1,78 %	45,95	- 0,53 %
α	-	0,3	0,299	- 0,16 %	0,306	+ 1,85 %	0,299	- 0,46 %
T	min	35	35,12	+ 0,12 %	36,0	+ 2,86 %	35,12	+ 0,35 %

Tabelle 3.12: Vergleich der aus der Simulation ermittelten Daten des Durchschnittskühlschranks

Wenn die Simulationsauflösung auf Minuten-Schritte erhöht wird, kommt es, aufgrund der größeren Zeitabstände, zu erhöhten Abweichungen. Diese Abweichungen des Durchschnittskühlschranks wirken sich direkt auf das Kollektiv von Kühlschränken mit statistisch verteilten Eigenschaften aus, da diese im Mittel auch die in Tabelle 3.12 angegebenen Abweichungen aufweisen.

In Abschnitt 2.2.3.3 wurde erläutert, wie mit Hilfe einer Anpassung der thermischen Parameter eine Verbesserung des Simulationsergebnisses erreicht werden kann. Um den Fehler beim vorliegenden Kühlschrankmodell zu reduzieren wird das (exakte) thermische Modell mit den um die bei der größeren Simulationsauflösung entstandenen Abweichungen reduziert und die thermischen Parameter neu berechnet:

Mit den neuen Werten ergibt sich $R_{th} = 0,551 \frac{\text{K}}{\text{W}}$ und $C_{th} = 13066,4 \frac{\text{J}}{\text{K}}$.

Mit diesen Werten wurde wiederum der Standardkühlschrank simuliert (blaue Kurve in Diagramm 3.6) und mit den erforderlichen Werten verglichen (Tabelle 3.12). Die Abweichungen von den Soll-Werten konnte deutlich reduziert werden. Bei einer Simulation eines Kollektivs von Kühlschränken mit gestreuten Parametern zeigte sich noch, dass eine Erhöhung des Wertes von R_{th} auf $0,56 \frac{\text{K}}{\text{W}}$ notwendig ist. Die so ermittelten Werte führen zu einer sehr guten Übereinstimmung mit dem gewünschten Ergebnis, dargestellt in Diagramm 3.8.

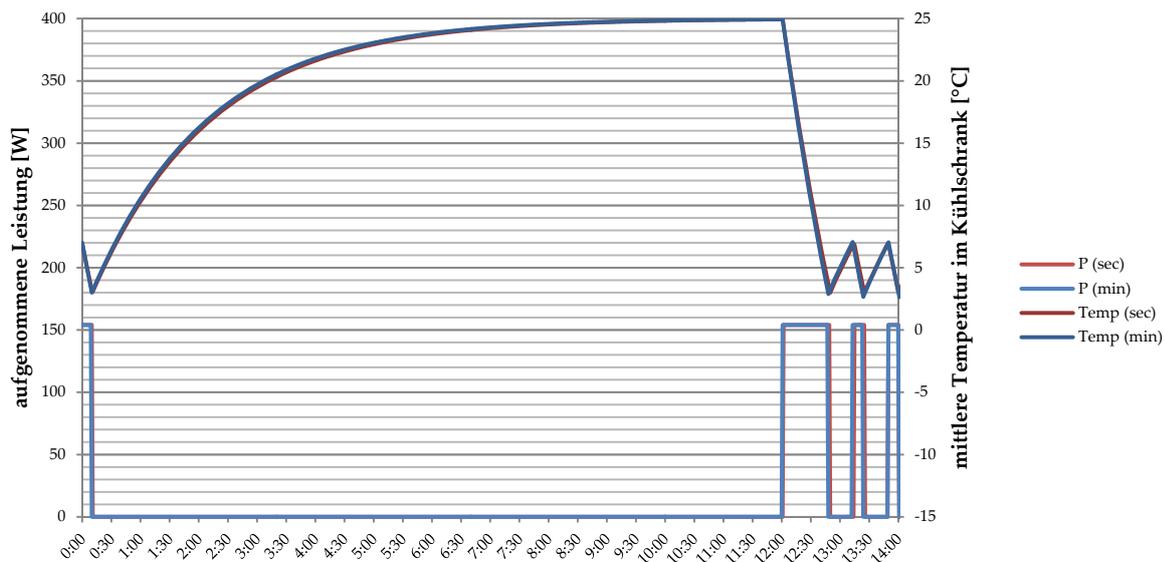


Diagramm 3.7: Vergleich der thermischen Parameter des Durchschnittskühlschranks für die verschiedenen Simulationsauflösungen. Parameter gem. Tabelle 3.12

In Diagramm 3.7 werden die verschiedenen thermischen Parameter anhand eines längeren Funktionsstillstandes zwischen 0:30 und 12:00 Uhr des Kühlschranks miteinander verglichen, um sicherzugehen, dass die thermische Reaktion des Systems die gleiche bleibt. Es ist gut zu erkennen, dass sich beide Simu-

lationsergebnisse sehr ähnlich sind. Sowohl die Temperaturerhöhung der Kühlschrankschranktemperatur als auch die Zeitspanne, die der Kühlschrank benötigt, um wieder seine Solltemperatur zu erreichen, weichen nicht signifikant voneinander ab.

Was bei dem gewählten thermischen Modell auffällt, ist die relativ rasche Erwärmung des Kühlschranks. In [18: S.25] wird im Mittel von einer Temperaturerhöhung von 0,3 bis $0,8 \frac{K}{W}$ (ohne Öffnen) ausgegangen. Würden diese Werte für die Parameterermittlung herangezogen, würde dies aber zu langen Periodendauern mit entsprechend verringerter relativer Einschaltdauer führen, da die dem System entzogene thermische Leistung und damit auch die elektrische Leistung der Kühlschränke gleich bleiben soll. Andererseits könnten auch die Schaltschwellen niedriger gewählt werden ($\vartheta_{SP} < 0,2^\circ C$). All dies widerspricht aber den Messungen in [9: S.50] bzw. Diagramm 3.5.

Man kann aber die schnellere Erwärmung des Kühlschranks durch das Öffnen des Kühlschranks und dem Einbringen von Lebensmitteln mit Zimmertemperatur erklären. Da von den Kühlschränken nur Durchschnittsdaten bezüglich der Leistungsaufnahme vorliegen, ist dementsprechend der normale Gebrauch des Kühlschranks (inkl. dem Öffnen) gemittelt in diesen Daten enthalten.

Wichtig für die Simulation mit DSM ist jedoch die Änderung der Leistungsaufnahme, die sich ergibt, wenn die Solltemperatur im Kühlschrank erhöht wird. In [1: S.63] wird das Potential an Leistungsreduktion ermittelt. So ergibt sich mit einer Temperaturerhöhung von z.B. $2^\circ C$:

$$\text{Reduktion} = 1 - \frac{\vartheta_A - \vartheta_{\text{Set, erhöht}}}{\vartheta_A - \vartheta_{\text{Set, normal}}} = 1 - \frac{25^\circ C - 7^\circ C}{25^\circ C - 5^\circ C} = 10\% \quad (3.13)$$

Diese kann direkt als Reduktion der elektrischen Leistung betrachtet werden, da beim Aufstellen des thermischen Modells alle thermischen Parameter auf die elektrische Seite bezogen wurden (Abschnitt 2.2.3.1). Bei einem Simulationsdurchlauf mit Kühlschränken von 25.000 Personen, die eine erhöhte Solltemperatur von im Mittel $2^\circ C$ aufweisen, ergibt sich eine Reduktion der durchschnittlichen Leistungsaufnahme von 9,23% (siehe Tabelle 3.14, letzte Spalte). Somit ist das thermische Modell, das hier für die Kühlschränke entwickelt wurde, für die Simulationsanforderungen ausreichend.

In Tabelle 3.13 sind die Parameter, die für die weiteren Simulationen verwendet werden, aufgelistet. Die Werte für die Standardabweichung der Parameterwerte wurden aus [9] entnommen.

Power_Nominal	154	W	20	%
Dir_therm_Flow	-1			
Efficiency	80	%	4	%
Switch_Point	4	$^\circ C$	20	%
Heat_Capacity	13066,4	J/K	20	%
Thermal_Res	0,56	K/W	20	%
Temp_Set	5	$^\circ C$	20	%
Temp_Ambiance	25	$^\circ C$	20	%
Start_Probability	40	%		
Operat_Sim_Start	30,5	%		

Tabelle 3.13: Parameter der Kühlschränke (Auszug aus Parameterdatei)

Mit diesem Parameterset wurden vier Simulationsdurchläufe mit einem Simulationszeitraum von 24h und einer Viertel-Stunden-Auflösung durchgeführt, die den Verbrauch von Kühlschränken für 25.000 Personen ermittelten. In Diagramm 3.8 sind die Ergebnisse zusammengefasst.

Es ist gut zu erkennen, dass die Leistungsaufnahme vieler Kühlschränke einen konstanten Charakter hat. Die Abweichungen von diesem konstanten Verlauf treten eher zufällig auf. Sie kommen durch die etwas geringe Anzahl an Geräten zustande, was sich im Rauschen in der Leistungsaufnahme zeigt. Je mehr Geräte simuliert werden, desto geringer wird dieses Rauschen.

Auch ist gut zu erkennen, dass mit der Wahl des Parameters `Operat_Sim_Start` es zu Beginn der Simulation kaum zu signifikanten Leistungsabweichungen kommt. Dieser Parameter entspricht ziemlich genau der relativen Einschaltdauer α von 0,3.

Tabelle 3.14 fasst weitere, während der Simulation ermittelte Daten zusammen. Zum einen ist gut zu erkennen, dass der Ausstattungsgrad kaum von der Vorgabe (40%) abweicht. Auch der Durchschnittsverbrauch über 24 Stunden weicht kaum vom zu erzielenden Wert von 18,6 W ab (abgesehen beim Simulationsdurchlauf mit geänderter Solltemperatur).

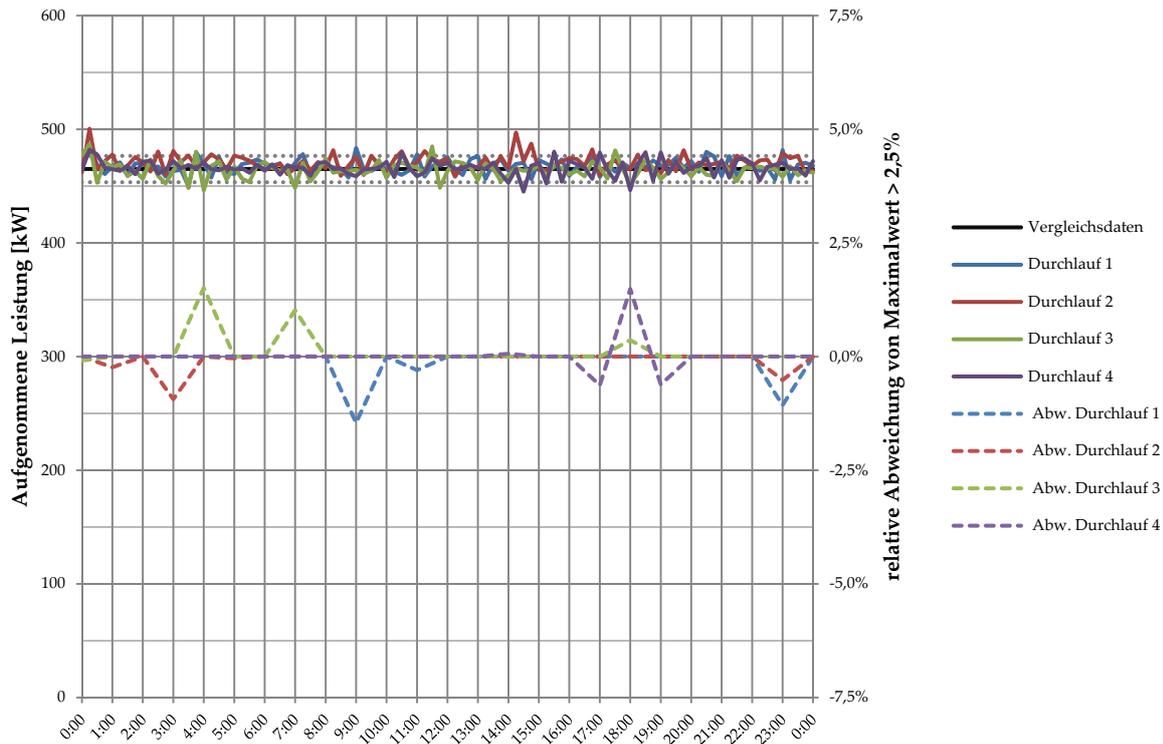


Diagramm 3.8: Ergebnis von vier Simulationsdurchläufen mit ca. 10.000 aktiven Kühlschränken mit Abweichungen von den Vergleichsdaten

	Durchlauf 1	Durchlauf 2	Durchlauf 3	Durchlauf 4	Temp_Set = 7°C
aktive Geräte	10.001	10.037	10.000	9.958	10072
Ausstattungsgrad	40,00 %	40,15 %	40,00 %	39,83 %	40,28 %
Durchschnittsverbrauch	18,70 W	18,83 W	18,57 W	18,64 W	16,88 W
Abweichung	0,53 %	1,22 %	-0,17 %	0,24 %	-9,23 %

Tabelle 3.14: Zusammenfassung von ermittelten Werten aus den Simulationsdurchläufen mit Kühlschränken von 25.000 Personen

3.5.3. Gefriergeräte

In [18: S.24] wird ein Ausstattungsgrad von Haushalten mit Gefriergeräten von 78% angegeben. Im Gegensatz zum Ausstattungsgrad mit Kühlschränken, der 98% beträgt.

Diese Angaben werden verwendet, um auf die Anzahl der Geräte pro Person schließen zu können:

$$\frac{78 \text{ Gefriergeräte} / 100 \text{ Haushalte}}{98 \text{ Kühlschränke} / 100 \text{ Haushalte}} \cdot 0,4 \frac{\text{Kühlschränke}}{\text{Person}} = 0,32 \frac{\text{Gefriergeräte}}{\text{Person}} \quad (3.14)$$

Mit dieser Angabe kann die mittlere Leistung eines Gefriergeräts angegeben werden (vgl. (3.9)):

$$\bar{P}_{\text{Gefriergerät}} = \frac{9,8 \text{ W}}{0,32} = 30,6 \text{ W} \quad (3.15)$$

Für Gefriergeräte ist in [18: S.24] die typische Einschaltdauer mit $\alpha = 0,2$ angegeben. Daraus lässt sich die durchschnittliche Anschlussleistung eines Gefriergeräts ableiten (vgl. (3.10)):

$$P_{\text{Gefriergerät}} = \frac{\bar{P}_{\text{Gefriergerät}}}{\alpha} = \frac{30,6 \text{ W}}{0,2} = 153 \text{ W} \quad (3.16)$$

Weiters gibt [18: S.24] die typische Periodendauer mit 60 bis 100 min an. Die Periodendauer wird für die Simulation mit 60 min angenommen. Die typische Solltemperatur für Gefriergeräte ist laut [1: S.66] -18°C. Die anderen Parameterwerte (Wirkungsgrad, Schalttemperatur und Umgebungstemperatur) werden gleich angenommen wie beim typischen Kühlschrank.

Mit diesen Daten werden nun die thermischen Parameter ermittelt (einsetzen in (3.7) und (3.8)):

$$R_{th} = \frac{1}{0,8 \cdot 153 \text{ W}} \cdot (25^\circ\text{C} - (-18)^\circ\text{C}) \cdot \frac{1}{0,2} = 1,757 \frac{\text{K}}{\text{W}} \quad (3.17)$$

$$C_{th} = 0,8 \cdot 153 \text{ W} \cdot \frac{0,2 \cdot (1-0,2) \cdot 60 \text{ min} \cdot 60 \text{ s} / \text{min}}{2 \text{ K}} = 35.251,2 \frac{\text{J}}{\text{K}} \quad (3.18)$$

Analog wie beim Durchschnittskühlschrank wird nun auch mit dem Durchschnittsgefriergerät verfahren und eine Simulation eines Gefriergeräts durchgeführt, einmal in Sekunden- und einmal in Minuten-Schritten, dargestellt in Diagramm 3.9 (rote und grüne Kurven).

Tabelle 3.15 gibt eine Zusammenfassung der nach der Simulation ermittelten Daten für einen Simulationszeitraum von 24h. Es ergibt sich ein ähnliches Bild wie bei den Kühlschränken. Bei einer feinen Auflösung liefert die Näherung des thermischen Modells eine gute Übereinstimmung mit den geforderten Werten. Dies kann man wiederum anhand der Temperaturverläufe in Diagramm 3.9 verifizieren.

	Einheit	Sollwert	sec-Auflösung	Abweichung von Sollwert	min-Auflösung	Abweichung von Sollwert	min-Auflösung (neue Werte)	Abweichung von Sollwert
R_{th}	K/W		1,757		1,757		1,819	
C_{th}	J/K		35.251,9		35.251,9		32.959,6	
$\bar{P}_{\text{Gefriergerät}}$	W	30,8	30,79	- 0,03 %	31,64	+ 2,73 %	30,68	- 0,37 %
α	-	0,2	0,201	+ 0,69 %	0,207	+ 3,47 %	0,201	+ 0,35 %
T	min	60	60,0	± 0,00 %	62,61	+ 4,35 %	60,0	± 0,00 %

Tabelle 3.15: Vergleich der aus der Simulation ermittelten Daten des Durchschnittsgefriergerätes

Die Abweichungen werden in das thermische Modell einbezogen und damit ergibt sich $R_{th} = 1,819 \frac{\text{K}}{\text{W}}$ und $C_{th} = 32.959,6 \frac{\text{J}}{\text{K}}$.

Eine Simulation mit diesen Werten in Minuten-Auflösung liefert deutlich bessere Werte (blaue Kurven in Diagramm 3.9 bzw. Einträge in Tabelle 3.15)

Diagramm 3.10 vergleicht die thermischen Parameter der verschiedenen Simulationsauflösungen wieder anhand eines längeren Inaktivitätsphase von 00:30 bis 12:00 Uhr. Auch hier ist die Abweichung der größeren zur feineren Auflösung nicht gravierend.

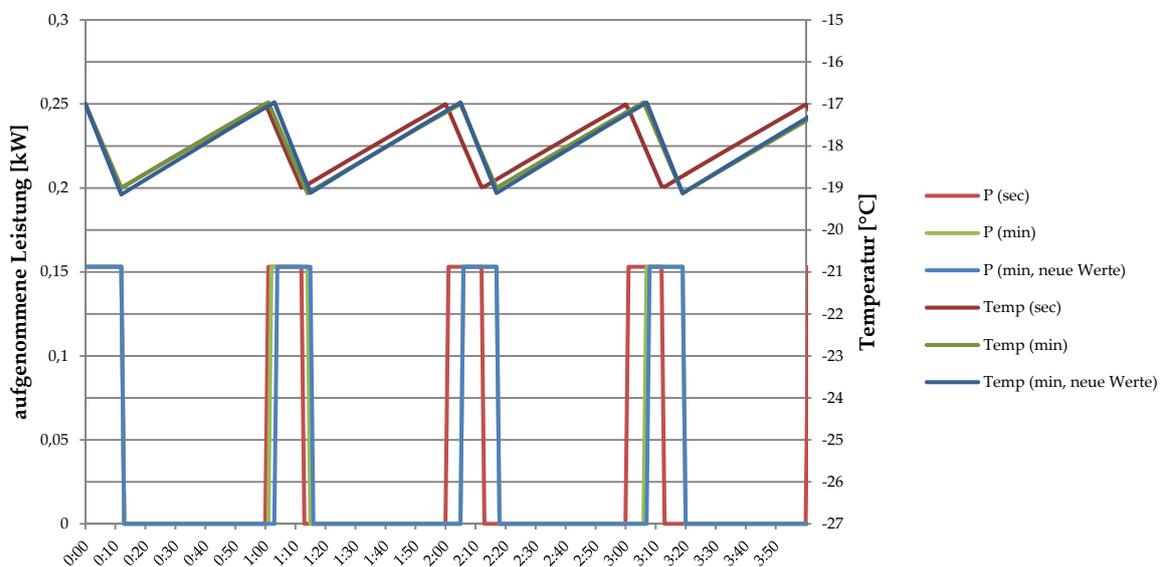


Diagramm 3.9: Vergleich der Auswirkungen der verschiedenen Parameterwerte und Berechnungsmethoden im thermischen Modell des Durchschnittsgefriergerätes; untere Kurven: Aufgenommene Leistung [kW]; obere Kurven: Temperatur im Kühlschrank [°C]

In [1: S.66] und [18: S.25] werden Ausfallszeiten von 14 bis 30 bei Gefrierschränken, sogar bis zu 62 Stunden bei Gefriertruhen angegeben, in denen diese Geräte ohne Stromversorgung eine Temperatur von -9°C nicht überschreiten. Beim vorliegenden thermischen Modell eines Durchschnittsgefriergerätes ist diese Zeit deutlich kürzer, nämlich nur 4 Stunden. Es gilt aber das gleiche wie bei den Kühlschränken, dass das normale Benutzen (d.h. öffnen und warme Lebensmittel einfrieren) im Mittel in diesen steileren Verlauf enthalten ist.

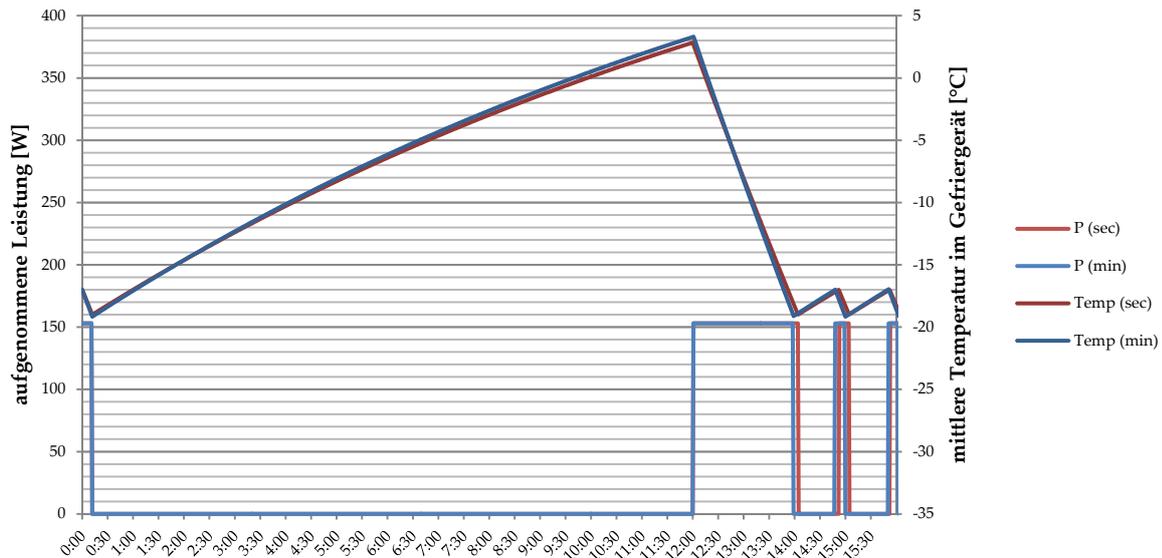


Diagramm 3.10: Vergleich der thermischen Parameter des Durchschnittskühlschranks für die verschiedenen Simulationsauflösungen. Parameter gem. Tabelle 3.15

Wie bei den Kühlschränken soll auch bei Gefriergeräten die Reduktion der aufgenommenen Leistung bei einer Solltemperaturerhöhung untersucht werden. In [1: S.69] wird angegeben, dass eine dauerhafte Temperaturerhöhung von 3°C bei Gefriergeräten möglich ist. Damit ergibt sich für die Reduktion aus Formel (3.13):

$$\text{Reduktion} = 1 - \frac{25^{\circ\text{C}} - (-15^{\circ\text{C}})}{25^{\circ\text{C}} - (-18^{\circ\text{C}})} = 7\% \quad (3.19)$$

Vergleicht man diesen Wert mit den Daten einer Simulation mit ca. 10.000 Gefriergeräten die eine um 3°C höhere Solltemperatur aufweisen (Tabelle 3.17, letzte Spalte) ergibt sich eine Reduktion von ca. 6%. Auch hier zeigt sich also eine gute Eignung des ermittelten thermischen Modells für die Simulation.

Power_Nominal	153	W	20	%
Dir_therm_Flow	-1			
Efficiency	80	%	4	%
Switch_Point	2	°C	20	%
Heat_Capacity	32959,6	J/K	20	%
Thermal_Res	1,819	K/W	20	%
Temp_Set	-18	°C	20	%
Temp_Ambiance	25	°C	20	%
Start_Probability	32	%		
Operat_Sim_Start	20,5	%		

Tabelle 3.16: Parameter der Gefriergeräte (Auszug aus Parameterdatei)

Tabelle 3.16 fasst die für die künftigen Simulationen verwendeten Parameterwerte zusammen. Die Breite der Streuungen wurde gleich wie bei den Kühlschränken gewählt, in Anlehnung an [9].

Mit diesen Werten wurden vier Simulationsläufe über 24h mit Gefriergeräten von 34.000 Personen durchgeführt. Die Ergebnisse sind in Diagramm 3.11 zusammengefasst. Es ergibt sich auch hier ein ähnliches Bild wie bei den Kühlschränken. Es wird annähernd konstant Leistung bezogen.

In Tabelle 3.17 sind weitere Daten aus der Simulation gegenübergestellt. Der erreichte Ausstattungsgrad sowie die durchschnittlich aufgenommene Leistung stimmen (abgesehen vom Simulationsdurchlauf mit erhöhter Solltemperatur) gut mit den geforderten Werten überein.

	Durchlauf 1	Durchlauf 2	Durchlauf 3	Durchlauf 4	Temp_Set = -15°C
aktive Geräte	10.821	10.908	10.848	10.825	10.842
Ausstattungsgrad	31,83 %	32,08 %	31,91 %	31,84 %	31,89 %
Durchschnittsverbrauch	9,86 W	9,98 W	9,88 W	9,82 W	9,20 W
Abweichung	0,62 %	1,81 %	0,82 %	0,24 %	-6,12 %

Tabelle 3.17: Zusammenfassung von ermittelten Werten aus den Simulationsdurchläufen mit Gefriergeräten von 34.000 Personen

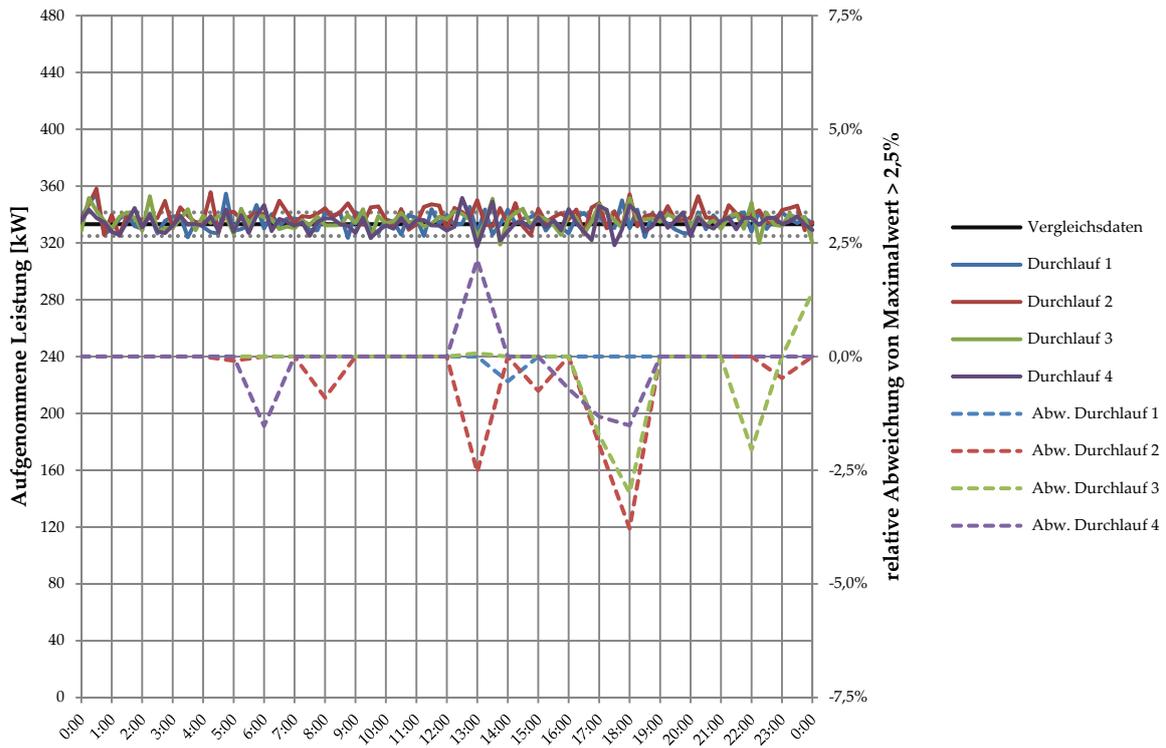


Diagramm 3.11: Ergebnis von vier Simulationsdurchläufen mit ca. 10.000 aktiven Gefriergeräten mit den Abweichungen von den Vergleichsdaten

3.6. Waschmaschinen

Waschmaschinen werden mit Hilfe der Klasse `Loadcurve_Operation` nachgebildet, da sie ein fixes Programm absolvieren, nachdem sie gestartet wurden. Um dieser Geräteklasse korrekt nachbilden zu können, müssen zuerst die Lastkurven, die durch das Waschprogramm bestimmt sind, ermittelt werden. Sind diese vorhanden können die Startzeiten sowie die Startwahrscheinlichkeiten ermittelt werden.

3.6.1. Lastkurven

Anhand von Messungen an einer Waschmaschine wurden die Lastgänge bei verschiedenen Programmen im Minutenraster ermittelt. In Diagramm 3.12 bis Diagramm 3.14 sind drei verschiedene Programme abgebildet.

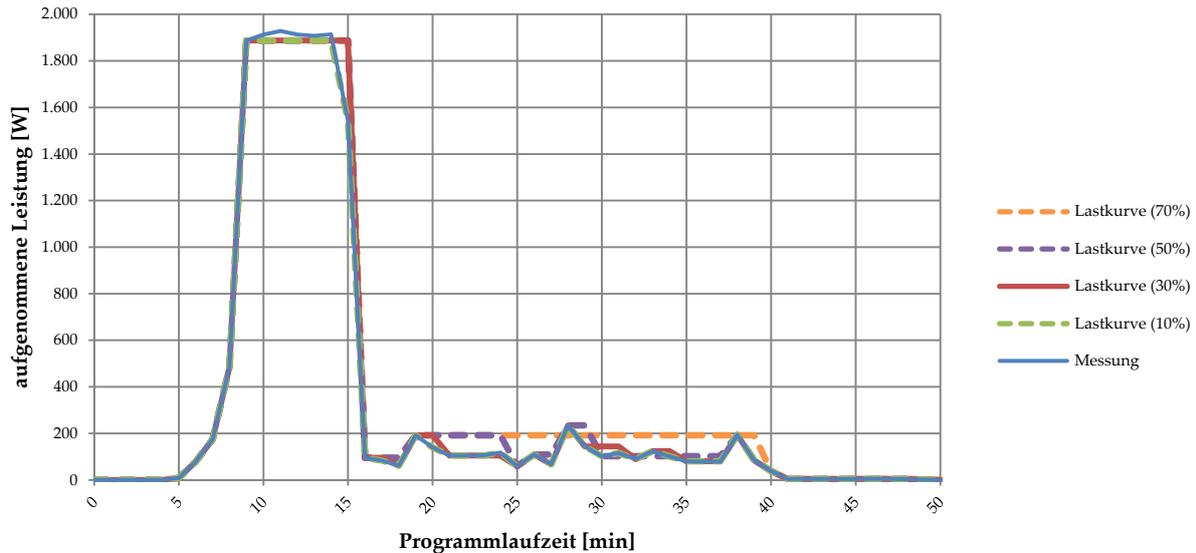


Diagramm 3.12: Lastgang einer Waschmaschine „AEG Lavamat 64618“ mit dem Programm „30min Super-schnell (30°)“ sowie daraus ermittelte Lastkurve (30% Abweichung von vorangegangenem Wert) und weitere mögliche Lastkurven (strichliert)

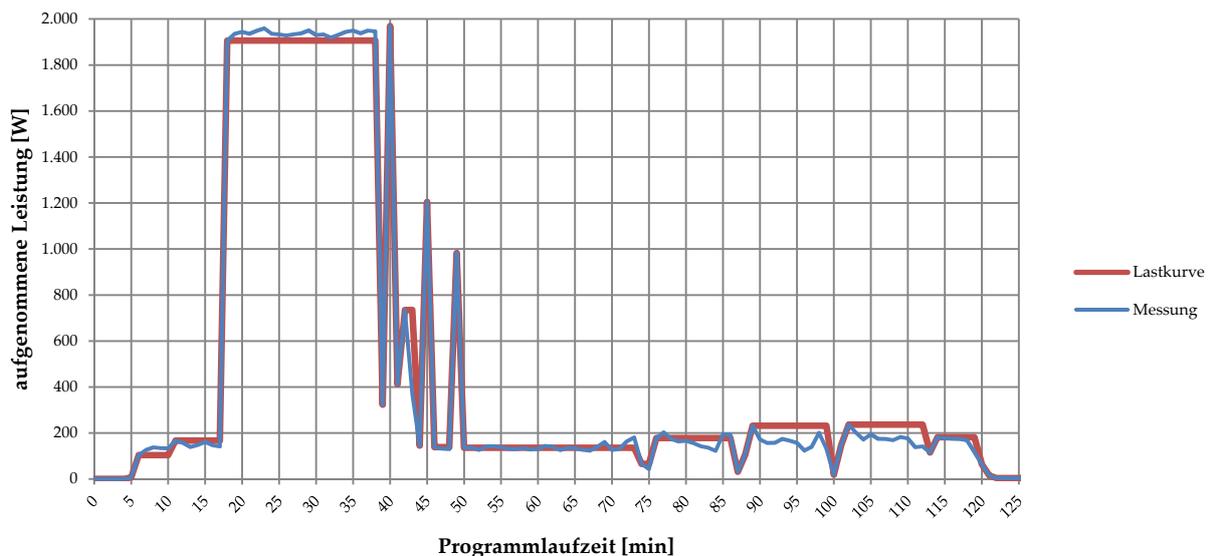


Diagramm 3.13: Lastgang einer Waschmaschine „AEG Lavamat 64618“ mit dem Programm „40°-60° Mix“ sowie daraus ermittelte Lastkurve (50% Abweichung von vorangegangenem Wert)

Um einerseits die Lastkurven so einfach wie möglich zu gestalten (siehe Abschnitt 2.2.7), und andererseits die Spezifika der gemessenen Waschmaschine nicht voll zu übernehmen, wurden die gemessenen Lastkurven für die Simulation adaptiert.

So wird von der Lastkurve nur dann ein neuer Wert von der Messung übernommen, wenn sich dieser um einen gewissen Prozentsatz zum vorhergehenden Wert unterscheidet. Je höher dieser notwendige Unterschied ist, desto größer wird die Wiedergabe der gemessenen Lastkurve. Der Wert wurde so gewählt, dass zum einen die gemessene Lastkurve gut nachgebildet wird (Diagramm 3.12), zum anderen nicht zu viele Punkte der Lastkurve für die Parameterdefinition übernommen werden müssen und diese so einfacher ist. In Diagramm 3.12 sind zusätzlich zur gewählten Lastkurve auch weitere mögliche Lastkurven eingezeichnet, die sich aus unterschiedlichen Unterscheidungsfaktoren ergeben.

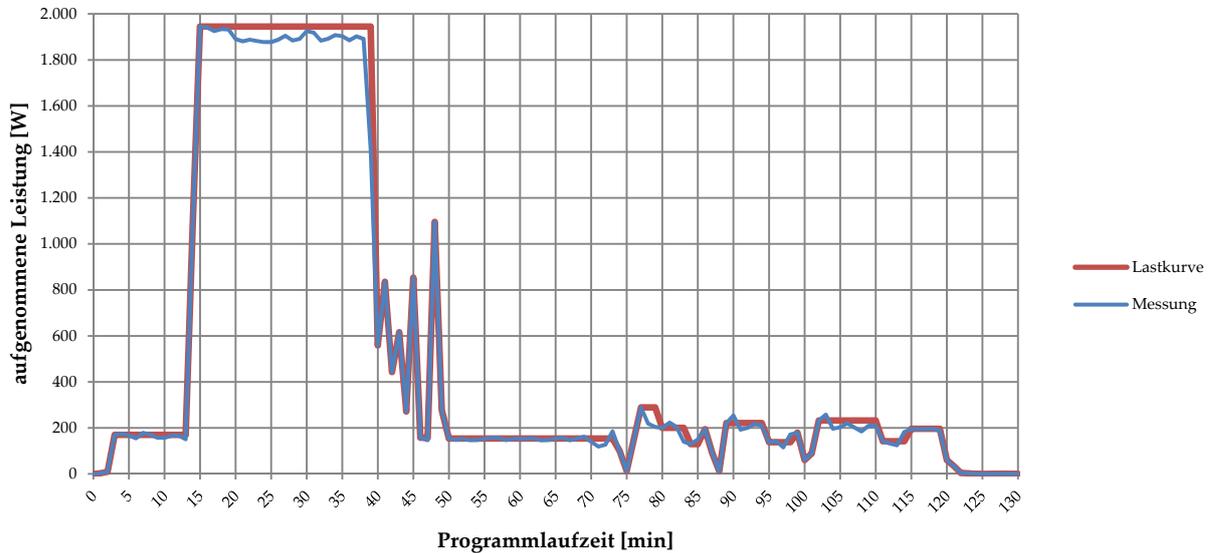


Diagramm 3.14: Lastgang einer Waschmaschine „AEG Lavamat 64618“ mit dem Programm „60°“ sowie daraus ermittelte Lastkurve (30% Abweichung von vorangegangenem Wert)

In den Lastkurven sind sehr schön die einzelnen Programmabschnitte einer Waschmaschine zu erkennen: zu Beginn die Wassereinlauf- und Heizphase, daran anschließen die Waschphase. Zum Ende hin sind klar die einzelnen Spülvorgänge sowie das Schleudern ersichtlich. Bei den Programmen „40°-60° Mix“ und „60°“ wird während des Waschvorganges anscheinend die Temperatur des Waschwassers ständig dem Sollwert nachgeführt, wodurch sich die kurzen Leistungsspitzen erklären.

3.6.2. Zusammenfassung der Parameterwerte

Für die Aufteilung der gesamten Waschvorgänge wurden folgende Werte angenommen:

30° Super schnell	40°-60° Mix	60°
30 %	60 %	10 %

Tabelle 3.18: Aufteilung der Programme auf alle Waschvorgänge von Waschmaschinen (Loadcurve_Allocation).

Diese Werte stellen nur eine Schätzung dar, da im Zuge der Recherche keine genauen Angaben über das Waschverhalten gefunden werden konnte.

Mit diesen Daten wurden dann die Einschaltzeiten mit ihren Streuungen sowie die Startwahrscheinlichkeit ermittelt.

Die Leistungswerte der Lastkurve werden mit 10%, die Dauer der jeweiligen Programmschritte mit 20% vom Mittelwert gestreut. Damit werden die verschiedenen Maschinen mit unterschiedlichen Beladungen und Programmen berücksichtigt. In Tabelle 3.19 sind die Parameterwerte für die Waschmaschinen zusammengefasst.

Als nicht unterbrechbare Teile der einzelnen Programme wurde jeweils die Heiz- und Waschphase sowie das abschließende Schleudern angegeben.

Time_Start	06:50	08:15	08:25	10:30	12:30	14:10	16:00	18:00	Uhr
'-> Std.Dev.:	45	75	45	30	40	90	30	60	min
Start_Probability	3,5	16	1,5	0,35	1	3	0,35	1,8	%
Power_Loadcurve	min	W	min	W	min	W			
	4	0,1	4	0,1	1	3,3			
	1	8,4	1	7,1	1	9,1			
	1	79,2	5	103,8	11	169,2			
	1	174,7	7	166,9	1	1104			
	1	487,2	21	1905,6	25	1944,6			
	7	1886,8	1	323	1	559,1			
	2	95,2	1	1968,8	1	835			
	1	61,5	1	411,5	1	443			
	2	191,9	2	734,7	1	615,8			
	4	104,7	1	145,3	1	271,9			
	1	59,1	1	1204,4	1	853,1			
	1	108,6	3	138	2	156,8			
	1	67,5	1	981,9	1	1095,6			
	1	234,4	24	135,4	1	276,6			
	3	143,6	2	65,8	24	153,8			
	1	89,7	11	177,6	1	99,9			
	2	123,5	1	31,2	1	12,8			
	3	80,2	1	105,6	1	152,4			
	1	193,3	11	231,4	3	289			
	1	84,4	1	18,6	4	200			
	1	38,5	1	147,6	2	129,4			
	8	3,9	11	235,8	1	194,5			
	1	1,3	1	115,4	1	91,9			
	1	0,1	6	181,6	1	12,2			
			1	64,4	6	221,2			
			1	15,5	4	137,9			
			24	3,9	1	179,6			
			1	1,8	1	60,7			
					1	89			
					9	231,9			
					4	141,5			
					5	196,1			
					1	60,2			
					1	33,7			
					1	3,9			
					1	2,6			
					7	0,1			
'-> Std.Dev.:	20	10	%						
Loadcurve_Allocation	30	60	10	%					
Loadcurve_non_stop_Parts	Start_idx	End_idx	Start_idx	End_idx	Start_idx	End_idx			
	5	11	5	15	4	17			
	18	20	20	28	29	37			

Tabelle 3.19: Parameter für die Waschmaschinen (Auszug aus Parameterdatei)

Bemerkenswert ist, dass die Startwahrscheinlichkeit für die Waschmaschinen keine hohen Werte erreicht, maximal 16% um 8:15 Uhr, das heißt, dass nur 16% der Waschmaschinen um diesen Zeitpunkt herum starten. Dies begründet sich einerseits mit der relativ geringen Anzahl von Geräten pro Kopf. Nach [1: S.87] ist der Ausstattungsgrad der Haushalte zwar 98%, pro Kopf ergibt das 0,47 Waschmaschinen pro Person. Andererseits sind nicht alle Waschmaschinen täglich im Einsatz [18: S.27], wodurch sich deren durchschnittliche Aktivität dementsprechend verringert.

Mit diesen Parametern wurden vier Simulationsdurchläufe mit den Waschmaschinen von 50.000 Personen und einem Simulationszeitraum von 24h durchgeführt.

Diagramm 3.15 fasst die Ergebnisse dieser Simulationsdurchläufe zusammen. Die Vergleichsdaten werden wiederum gut nachgebildet, bis auf größere Abweichungen bei der relativ schmalen Spitze um 9:30 Uhr. Wie man erkennen kann, sind die einzelnen Abweichungen nicht miteinander in Korrelation, weil sie zufällig durch die erzeugten Parameterinstanzen auftreten. Wird eine höhere Anzahl an Geräten simuliert, werden auch diese Abweichungen geringer.

Der restliche Tagesverlauf wird sehr gut nachgebildet. Auch der Durchschnittsverbrauch liegt nahe dem in Tabelle 3.1 geforderten Wert von 9,2 W pro Person.

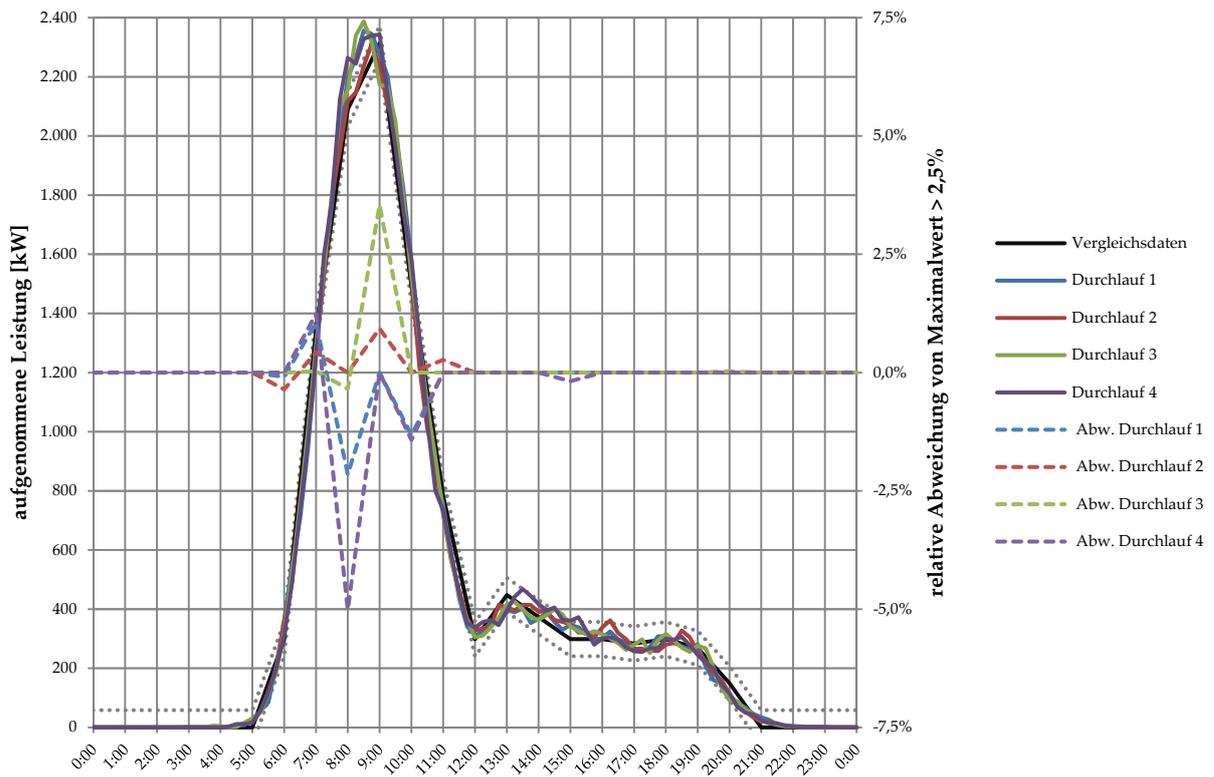


Diagramm 3.15: Ergebnis von vier Simulationsdurchläufen mit ca. 12.000 aktiven Waschmaschinen mit den Abweichungen von den Vergleichsdaten

	Durchlauf 1	Durchlauf 2	Durchlauf 3	Durchlauf 4
aktive Geräte	12.589	12.561	12.724	12.565
Ausstattungsgrad	25,18 %	25,12 %	25,45 %	25,13 %
Durchschnittsverbrauch	9,11 W	9,04 W	9,18 W	9,18 W
Abweichung	- 1,03 %	- 1,71 %	-0,19 %	-0,17 %

Tabelle 3.20: Zusammenfassung von ermittelten Werten aus den Simulationsdurchläufen mit Waschmaschinen von 50.000 Personen

3.7. Geschirrspüler

3.7.1. Lastkurve

Bei der Lastkurve für Geschirrspüler wurde ähnlich verfahren wie bei den Waschmaschinen, mit dem einzigen Unterschied, dass für die Geräteklasse Geschirrspüler nur ein einziger Lastgang herangezogen wird. Dieser ist in Diagramm 3.16 dargestellt.

Auch hier sind die einzelnen Programmabschnitte deutlich zu erkennen: Wassereinlauf, Vorspülen, Abpumpen, Aufheizen des Spülwasser und Spülen, Abpumpen, Aufheizen des Wasser für Klarspülen und Klarspülen, Abpumpen (ca. bei Minute 100), Trocknen und erneutes Abpumpen (ca. bei Minute 140).

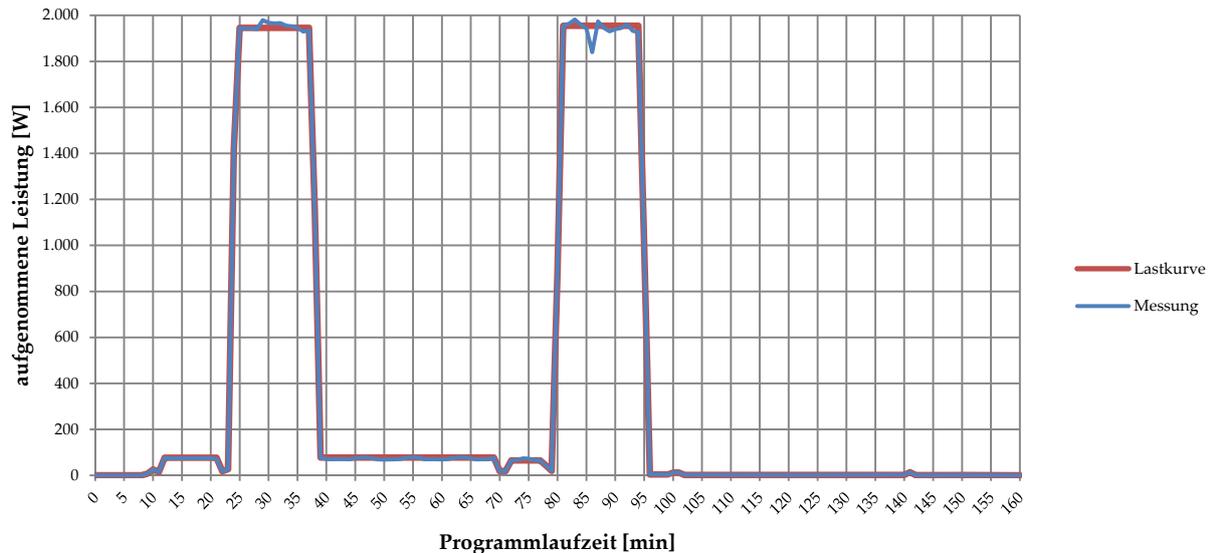


Diagramm 3.16: Lastgang eines Geschirrspülers „Whirlpool ADG8527“ mit dem Programm „60°“ sowie daraus ermittelte Lastkurve (30% Abweichung von vorangegangenen Wert)

3.7.2. Zusammenfassung der Parameterwerte

Mit Hilfe der gewählten Lastkurve wurden die zeitlichen Parameter für diese Geräteklasse ermittelt.

Als nicht unterbrechbare Teile der Lastkurve wurde das Aufheizen des Spülwassers inklusive Spülvorgang sowie das Aufheizen des Wassers für das Klarspülen inklusive Klarspülen definiert.

Wiederrum bemerkenswert sind die, in den Parameterwerten in Tabelle 3.21 ersichtlichen, geringen Einschaltwahrscheinlichkeiten (maximal 9% um 12:10 Uhr). Diese lassen sich wieder, wie bei den Waschmaschinen, mit der relativ geringen Anzahl von Maschinen pro Person erklären. Laut [1: S.91] gaben im Jahr 2000 51% der Haushalte an, einen Geschirrspüler zu besitzen, woraus sich 0,25 Geräte pro Person ergeben.

Mit den gewählten Parametern wurden vier Simulationsdurchläufe mit Geschirrspülern von ca. 50.000 Personen durchgeführt. Die Ergebnisse sind in Diagramm 3.17 zusammengefasst.

Das Simulationsergebnis weicht hier bei der Lastspitze um 13:00 deutlich von den Vergleichsdaten ab. Mit dem gewählten Lastgang konnte auch keine bessere Übereinstimmung erzielt werden. Da der gewählte Lastgang aus zwei Leistungsspitzen besteht, die ca. 70 min auseinander liegen, kann keine einzelne Lastspitze auftreten, selbst wenn viele Geschirrspüler gleichzeitig starten würden. Es würde immer eine Doppelspitze entstehen. Im vorliegenden Ergebnis zeigt sich dieser Umstand in einer verbreiterten Lastspitze.

Dadurch kommt es zu den hohen Abweichungen um 13:00 Uhr (bis zu 10% zu geringe Leistungsaufnahme) und um 14:00 Uhr (bis zu 19% erhöhte Leistungsaufnahme bezogen auf den Maximalwert der Vergleichsdaten), welche stark miteinander korreliert sind.

Der restliche Tagesverlauf weist hingegen kaum Abweichungen auf, weshalb auch für die weiteren Simulationen der gewählte Parametersatz verwendet wird. Für eine Verbesserung der Übereinstimmung müssten noch weitere Lastkurven in den Parametersatz aufgenommen und die Startzeiten angepasst werden. Davon wurde aber im Rahmen dieser Arbeit abgesehen.

Time_Start	05:30	07:45	10:30	12:10	12:20	14:00	17:45	19:30	Uhr
'-> Std.Dev.:	35	40	20	40	25	90	40	40	min
Start_Probability	0,75	1,5	1,5	9	4	1	5	1,2	%
Power_Loadcurve	min	W							
	1	7,4							
	1	25,7							
	1	14,8							
	10	76,8							
	1	17,5							
	1	27,2							
	1	1420,7							
	13	1945,2							
	1	1154,7							
	31	77,3							
	2	18,3							
	6	65,1							
	1	43,2							
	1	19,5							
	1	869,2							
	14	1954,3							
	1	1011,9							
	4	3,7							
	2	11,8							
	39	1,8							
	1	14,2							
	17	0,9							
'-> Std.Dev.:	20	10							%
Loadcurve_non_stop_Parts	Start_idx	End_idx							
	15	18							
	7	11							

Tabelle 3.21: Parameter für die Geschirrspüler (Auszug aus Parameterdatei)

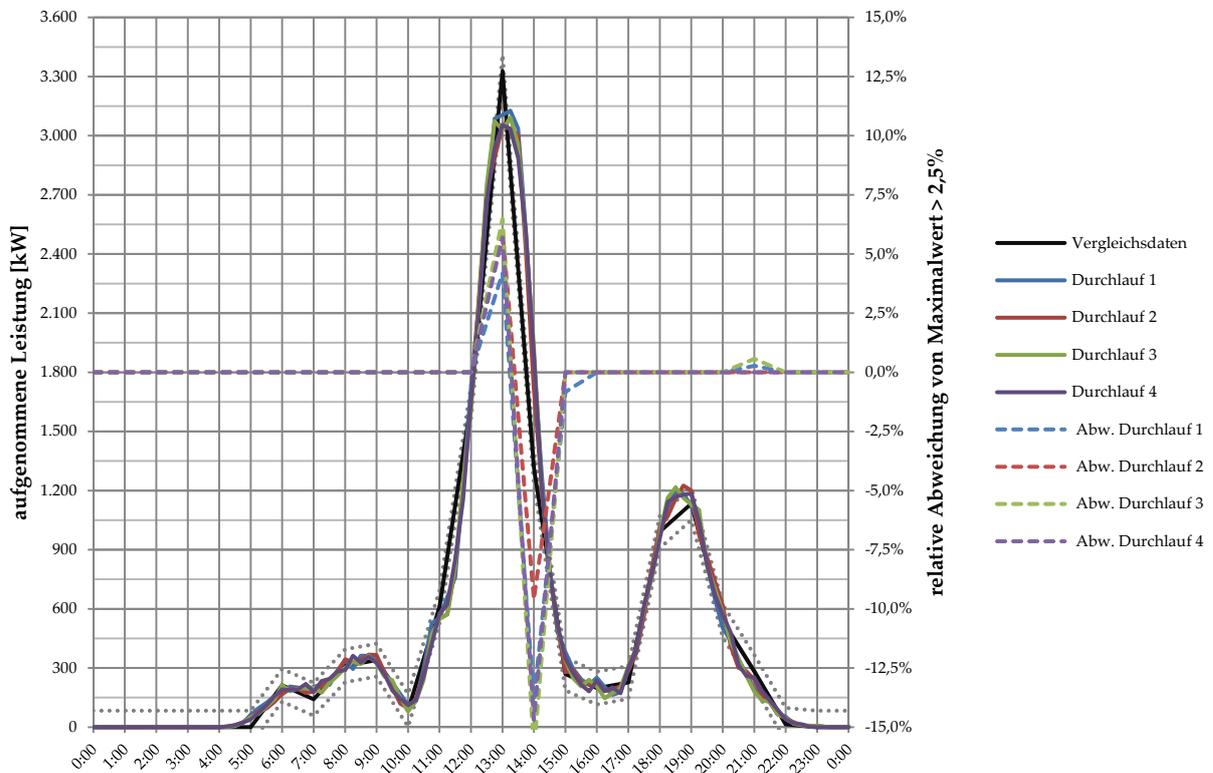


Diagramm 3.17: Ergebnis von vier Simulationdurchläufen mit ca. 10.000 aktiven Waschmaschinen mit den Abweichungen von den Vergleichsdaten

	Durchlauf 1	Durchlauf 2	Durchlauf 3	Durchlauf 4
aktive Geräte	10.970	10.755	10.826	10.787
Ausstattungsgrad	21,94 %	21,51 %	21,65 %	21,57 %
Durchschnittsverbrauch	10,09 W	9,91 W	9,96 W	9,95 W
Abweichung	+ 4,07 %	+ 2,13 %	+ 2,71 %	+ 2,54 %

Tabelle 3.22: Zusammenfassung von ermittelten Werten aus den Simulationsdurchläufen mit Waschmaschinen von 50.000 Personen

Tabelle 3.22 fasst noch weitere, während der Simulation ermittelte Daten zusammen. Der Ausstattungsgrad der aus den aktiven Geräten ermittelt wird, beträgt hier ca. 22%. Dieser Wert liegt im Bereich der Anzahl der Geräte pro Kopf von 0,25. Daraus lässt sich ableiten, dass eine Mehrheit der Geschirrspüler im Modell täglich im Einsatz ist.

Die großen Abweichungen im Tagesverlauf von den Vergleichsdaten äußern sich auch im Durchschnittsverbrauch, der deutlich höher ausfällt. Gemäß Tabelle 3.1 sollte ein Wert von 9,7 W pro Person erreicht werden.

3.8. Gesamtverbrauch

Diagramm 3.18 zeigt zusammenfassend die Zusammensetzung der Gesamtleistung aus den einzelnen Verbrauchergruppen, die in den vorhergehenden Abschnitten definiert wurden. Dazu wurde eine Simulation der Geräte von 50.000 Personen über 24h mit einer Auflösung von 5 Minuten durchgeführt.

Der Verlauf der Gesamtleistung entspricht nicht ganz dem Verlauf des H0-Standard-Lastprofils, da noch einige Verbrauchergruppen in der Simulation nicht berücksichtigt wurden (z.B. Wäschetrockner, Umwälzpumpen, die Bereiche „Kochen“ und „Warmwasser“).

Mit den vorhandenen Geräteklassen ist aber dementsprechende Erweiterung leicht möglich, wurde aber im Rahmen dieser Arbeit nicht durchgeführt.

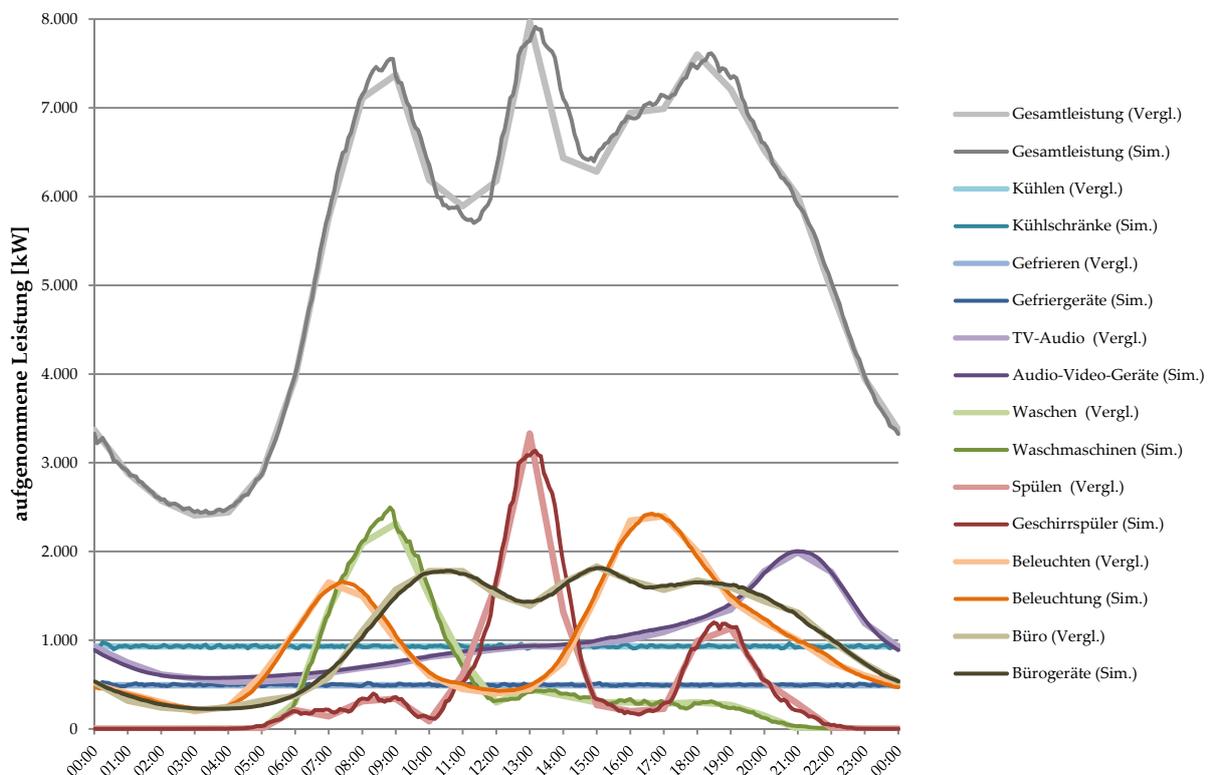


Diagramm 3.18: Ergebnis der Simulation der Leistungsaufnahme von Verbrauchern von 50.000 Personen

Tabelle 3.23 fasst weitere, bei der Simulation ermittelte Daten zusammen. Diese zeigen ein sehr ähnliches Bild wie jene Daten der vorhergehenden Abschnitte, die jeweils von ca. 10.000 Geräten stammen. Die Abweichungen des Durchschnittsverbrauchs bewegen sich durchaus in einem engen Rahmen in der Nähe der Vergleichsdaten von Tabelle 3.1. Nur die Abweichung bei den Geschirrspülern fällt höher aus, aus

den bereits erläuterten Gründen. Die Abweichung des gesamten Durchschnittsverbrauch von dem in Tabelle 3.1 angegebenen Wert von 108,2 W pro Person ist dennoch ausreichend gering.

	Gesamt	Bürogeräte	Audio-Video-Geräte	Beleuchtung	Kühlschränke	Gefriergeräte	Waschmaschinen	Geschirrspüler
aktive Geräte	197.952	48.909	40.067	49.478	19.894	16.022	12.704	10.878
Ausstattungsgrad	-	97,8 %	80,1 %	99,0 %	39,8 %	32,0 %	25,4 %	21,8 %
Durchschnittsverbrauch	109,2 W	22,0 W	19,8 W	19,6 W	18,6 W	9,9 W	9,2 W	10,1 W
Abweichung	+ 0,90 %	- 0,11 %	+ 1,81 %	+ 0,85 %	- 0,05 %	+ 1,15 %	+ 0,15 %	+ 3,74 %

Tabelle 3.23: Zusammenfassung von ermittelten Werten aus einem Simulationsdurchlauf mit Geräten von 50.000 Personen

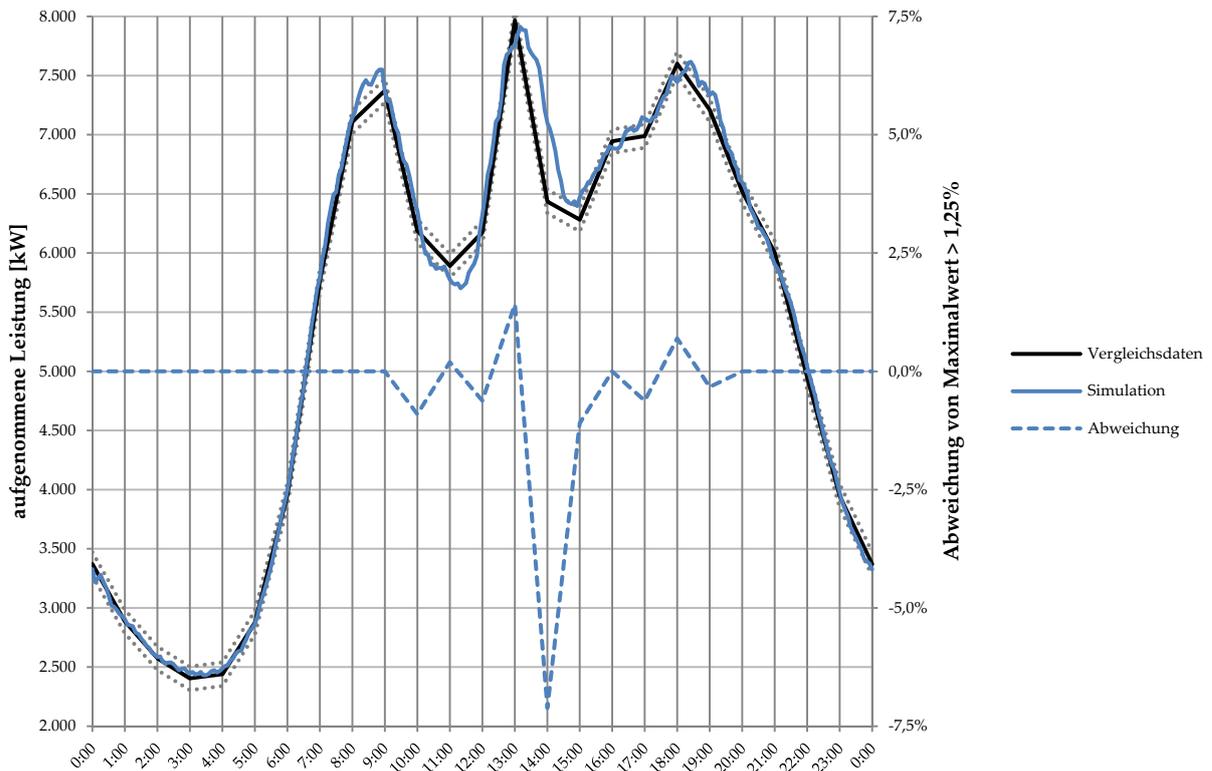


Diagramm 3.19: Abweichung des simulierten Gesamtverbrauchs von den Vergleichsdaten

Interessant ist auch die Abweichung des Tagesverlaufs des Gesamtverbrauchs. Dieser wird in Diagramm 3.19 dargestellt. Wie nicht anders zu erwarten ergibt sich eine große Abweichung um 14:00 Uhr von fast 8,5% vom Maximalwert der Vergleichsdaten. Diese wird durch die Geschirrspüler verursacht. Die Abweichungen des restlichen Tagesverlaufs befinden sich im Bereich von $\pm 2,5\%$ vom Maximalwert, in der Zeit von 20:00 bis 9:00 Uhr sogar unter $\pm 1,25\%$.

Es liefert also der hier ermittelte Satz an Gerätegruppen mit deren Parameterwerten eine relativ gute Nachbildung des gewünschten Verlaufes. Dieser Gerätesatz bildet nun die Basis für alle weiteren Vergleiche.

4. Potentiale der verwendeten Gerätegruppen

In [1] wurden bereits Überlegungen und Untersuchungen zu den Potentialen der verschiedenen Gerätegruppen für DSM gemacht. Diese bilden auch weitgehend die Grundlage für die Annahmen der möglichen Leistungsreduzierungen für die einzelnen Gerätegruppen bei den ersten Simulationen (siehe Kapitel 5).

Im Folgenden sollen kurz diese Potentiale vorgestellt werden. Weiters soll auch kurz erläutert werden, wie diese für die einzelnen Gerätegruppe als DSM-Output definiert werden.

4.1. Bürogeräte

Laut [1: S.32] kann bei allen Geräten dieses Sektors der Stand-by-Verbrauch abgeschaltet werden, ausgenommen bei Tintenstrahldruckern. Für Desktop-PCs, Monitore und Notebooks wird eine Leistungsreduktion von 30% für möglich erachtet.

Gerätetyp	Geräte pro Person	Anteil
Desktop PCs	0,12	61,6 %
Notebooks	0,27	
Monitore	0,30	
Drucker (Tinte)	0,23	21,0 %
Drucker (Laser)	0,18	19,4 %
Sonstiges	0,02	

Tabelle 4.1: Aufteilung der Geräte im Sektor Büro

Mit den Daten aus Tabelle 3.2 lassen sich die Anteile der einzelnen Gerätegruppen am Gesamtsektor ermitteln (dargestellt in Tabelle 4.1). Daraus geht nun hervor, dass ca. 60% der Geräte eine Leistungsreduktion durchführen können, ca. 20% ihren Stand-by-Verbrauch ausschalten können und 20% keinen Beitrag zur Leistungsreduktion liefern (Tintenstrahldrucker). Mit diesen Daten kommt man zu folgender Aufteilung der DSM-Output-Moden:

DSM_Output_Mode					
+--->	No_DSM_Function		-->	20	%
+--->	Turn_Off		-->	0	%
+--->	Turn_Off_Stand_by		-->	20	%
'--->	Reduce_Input_Power		-->	60	%
Power_Reduction	30 %		10 %		

Tabelle 4.2: Einstellungen für den DSM-Output der Bürogeräte (Auszug aus Parameterdatei)

Nicht alle Geräte haben den gleichen Anteil an der Gesamtleistung, wodurch sich durch die vereinfachende Zusammenfassung eine Unsicherheit auch in Bezug auf diese Aufteilung der Möglichkeit der Leistungsreduktionen ergibt. Um hier für ein besseres Ergebnis zu sorgen, könnte der gesamte Sektor auf die hier vorgestellten Gerätegruppen aufgeteilt werden.

Bei einer hohen Personenanzahl sollte aber bei der Simulation ein repräsentatives Ergebnis zustande kommen.

4.2. Audio-Video-Geräte

In [1: S.48] wird von einem Reduzierungspotential von 20% bei Fernsehern und 5% bei HiFi-Geräten ausgegangen. Die restlichen Geräte könnten über ihren Stand-by-Verbrauch zu der Reduzierung beitragen.

Ähnlich wie bei den Bürogeräten werden die einzelnen Potentiale gemäß des Anteils an der Gesamtanzahl der Geräte des Sektors aufgeteilt, dargestellt in Tabelle 4.3. Über die Anteile von TV-Display Total und Hi-Fi mit Radio ergibt sich für diese Geräte eine mittlere Leistungsreduktion von ca. 12%. Diese Reduktion können ca. 50% der Geräte bewerkstelligen, die restlichen 50% können ihren Stand-by-Verbrauch reduzieren.

Gerätetyp	Geräte pro Person	Anteil	
TV-Display Total	0,53	22,3%	
Hi-Fi	0,22	29,8 %	52,1 %
Radio	0,49		
Video Total	0,41		
Videogames Konsolen	0,35	47,9 %	
Set Top Boxen	0,38		

Tabelle 4.3: Aufteilung der Geräte im Sektor Audio-Video

Damit ergibt sich für die Aufteilung des DSM-Output zu:

DSM_Output_Mode					
+--->	No_DSM_Function		-->	0	%
+--->	Turn_Off		-->	0	%
+--->	Turn_Off_Stand_by		-->	50	%
'--->	Reduce_Input_Power		-->	50	%
Power_Reduction	12	%	20	%	

Tabelle 4.4: Einstellungen für den DSM-Output der Bürogeräte (Auszug aus Parameterdatei)

Wie bereits bei den Bürogeräten erwähnt, könnte auch dieser Sektor für ein besseres Ergebnis auf einzelne Gerätegruppen aufgeteilt werden.

4.3. Beleuchtung

Für den Sektor Beleuchtung wird in erster Linie nur von einer Helligkeitsreduktion ausgegangen. In [1: S.58] wird hier von einem Potential von 9% Wirkleistungseinsparung verwendet. Dieser Wert stellt eine kaum bemerkbare Beeinträchtigung der Nutzer dar.

DSM_Output_Mode					
+--->	No_DSM_Function		-->	0	%
+--->	Turn_Off		-->	0	%
'--->	Reduce_Input_Power		-->	100	%
Power_Reduction	10	%	20	%	

Tabelle 4.5: Einstellungen für den DSM-Output für die Beleuchtung (Auszug aus Parameterdatei)

Es könnte aber noch ein Anteil an Beleuchtungskörpern betrachtet werden, die sich bei ungünstigen Netzsituationen abschalten und die verbleibenden Beleuchtungskörper eine Art „Notbeleuchtung“ bilden. Diese Situation wird in den in dieser Arbeit durchgeführten Simulationen aber nicht genauer betrachtet.

4.4. Kühlschränke und Gefriergeräte

Bei Kühlschränken und Gefriergeräten wird eine Solltemperaturänderung zur Leistungsbeeinflussung durchgeführt. Dabei wird die Temperatur nur soweit erhöht, dass diese über einen längeren Zeitraum zu keiner Beeinträchtigung der Haltbarkeit der Lebensmittel führt.

In [1: S.62] wird hier eine Temperaturerhöhung von 2°C bei Kühlschränken, und 3°C bei Gefriergeräten angegeben, wobei für letzteren Wert keine zeitliche Einschränkung angegeben wird. Bei einem Stromausfall kann die Temperatur in einem Gefriergerät um bis zu 9°C ansteigen (siehe Abschnitt 3.5.3). Um bei Gefriergeräten ein höheres Potential für DSM nutzen zu können, wird daher mit einer Solltemperaturänderung von 5°C gearbeitet wobei jedoch darauf zu achten ist, dass diese Temperaturerhöhung nicht für einen allzu langen Zeitraum vorherrscht.

4.5. Waschmaschinen und Geschirrspüler

Für Waschmaschinen und Geschirrspüler kann kein konkretes Einsparungspotential angegeben oder festgestellt werden, da die mögliche Leistungsreduktion sehr von der Art der Verschiebung des Programmes sowie der Tageszeit abhängt.

5. Simulation von Verbrauchern mit DSM

Das in Kapitel 3 ermittelte Modell von verschiedenen elektrischen Verbrauchern im Haushalt und die im Kapitel 2 entwickelten Möglichkeiten zur Leistungsbeeinflussung dienen nun als Ausgangspunkt um anhand verschiedener Szenarien erste Untersuchungen zu der Leistungsfähigkeit eines möglichst einfachen DSM-Systems durchführen zu können. Dabei wird auf die in Kapitel 4 angegebenen Potentiale Rücksicht genommen.

Allen Szenarien liegt eine Netzfrequenz zugrunde, die von außen dem Simulationsmodell vorgegeben und vom Verhalten der Verbraucher nicht beeinflusst wird (siehe Abschnitt 2.4).

Mit Hilfe der verschiedenen Szenarien wird nun versucht, Anforderungen an das DSM-System zu ermitteln, um ein günstiges Verhalten der Verbraucher in Bezug auf die Leistungsaufnahme zu erzielen.

5.1. Verschiebung Lastspitzen

In diesem ersten Szenario wird angenommen, dass über die Netzfrequenz versucht wird, die drei Leistungsspitzen, die sich beim Betrieb der betrachteten Geräte ergeben, auf andere Zeitpunkte zu verschieben. Der für diesen Zweck angenommene Frequenzverlauf kann Diagramm 5.1 entnommen werden.

Die Änderung der Netzfrequenz wird hier mit einem relativ hohen Wert von einem Hertz angenommen, der sich aber nur mit einer Rate von 1 Hz pro Stunde ändert.

Als Verbraucherkollektiv werden die Geräte von 5.000 Personen herangezogen. In Diagramm 5.1 ist die Leistungsaufnahme dieser Geräte ohne Beeinflussung durch DSM dargestellt. Auch wird darin das Simulationsergebnis den Vergleichsdaten gegenübergestellt. Anhand des dargestellten Frequenzverlaufs ist gut zu erkennen, dass bei den auftretenden Lastspitzen die Netzfrequenz reduziert wird.

Diese Lastspitzen entsprechen nicht den Lastspitzen der gemittelten Tageslastganglinie (Diagramm 3.1). Für eine Beeinflussung dieser müsste ein anderer Frequenzverlauf gewählt werden. Da aber nicht alle Sektoren für diese Arbeit implementiert wurden, werden nur die hier vorgestellten und deren Leistungsaufnahme betrachtet.

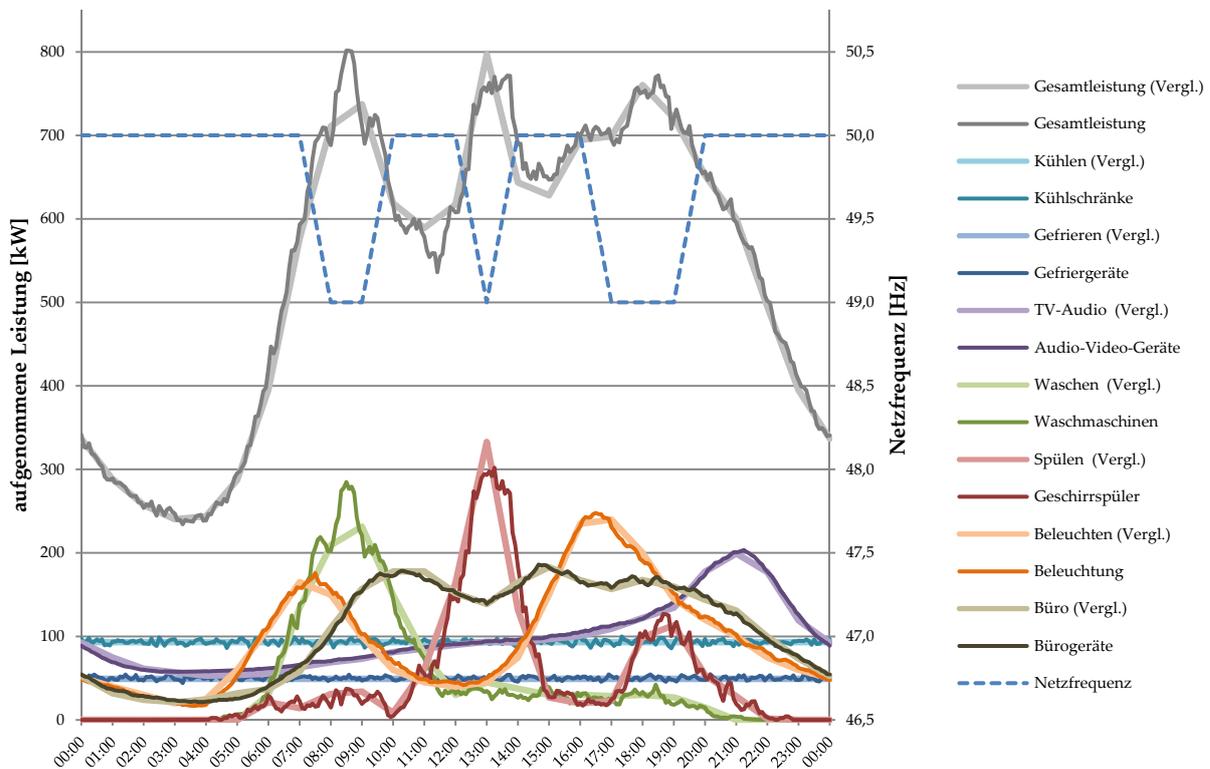


Diagramm 5.1: Unbeeinflusste Leistungsaufnahme des für die folgenden Untersuchungen verwendeten Verbraucherkollektivs mit den Geräten von 5.000 Personen.

Als mittlere Frequenzschwelle für alle DSM- Einheiten wurde für dieses Szenario eine Abweichung von 0,5 Hz festgelegt (DSM-Input `Frequency_Response_Simple`).

Die Potentiale für die Leistungsreduktion wurden gemäß Kapitel 4 für die einzelnen Sektoren angenommen.

Für Waschmaschinen und Geschirrspüler wurde zunächst angenommen, dass alle Geräte, die sich an der Leistungsbeeinflussung beteiligen, dies durch eine Verschiebung ihrer Startzeit durchführen (DSM-Output `Postpone_Start`). Die Startzeit wird im Mittel um 30 min verschoben (`Time_Postpone_Start`) jedoch im Mittel nicht länger als 120 min (`Time_Postpone_max`). Diese Zeitwerte werden mit einer sehr großen Standardabweichung von 30% des Mittelwerts gestreut.

5.1.1. Einfluss der Standardabweichung der Frequenzschwelle

Als erste Untersuchung wird der Einfluss der Standardabweichung der Frequenzschwelle auf das Simulationsergebnis betrachtet. Diese stellt, wie bereits im Abschnitt 2.3.1.2 ausgeführt, einerseits die unterschiedliche Messgenauigkeit der DSM-Einheiten dar, andererseits könnte durch eine bei der Produktion bewusst erfolgte Streuung der Frequenzschwellen versucht werden, gleichzeitige Leistungsänderungen der Verbraucher zu vermeiden. Welcher dieser beiden Effekte hauptsächlich für die Streuung der Frequenzschwelle verantwortlich ist, ist an dieser Stelle nicht von großer Bedeutung.

Es wurden nun verschiedene Simulationsdurchläufe mit unterschiedlichen Standardabweichungen der Frequenzschwelle `Frequency_Level` durchgeführt. Dabei wurde angenommen, dass sich alle Geräte an der Leistungsreduktion beteiligen. Die Werte wurden mit einer Auflösung von 5 Minuten ermittelt.

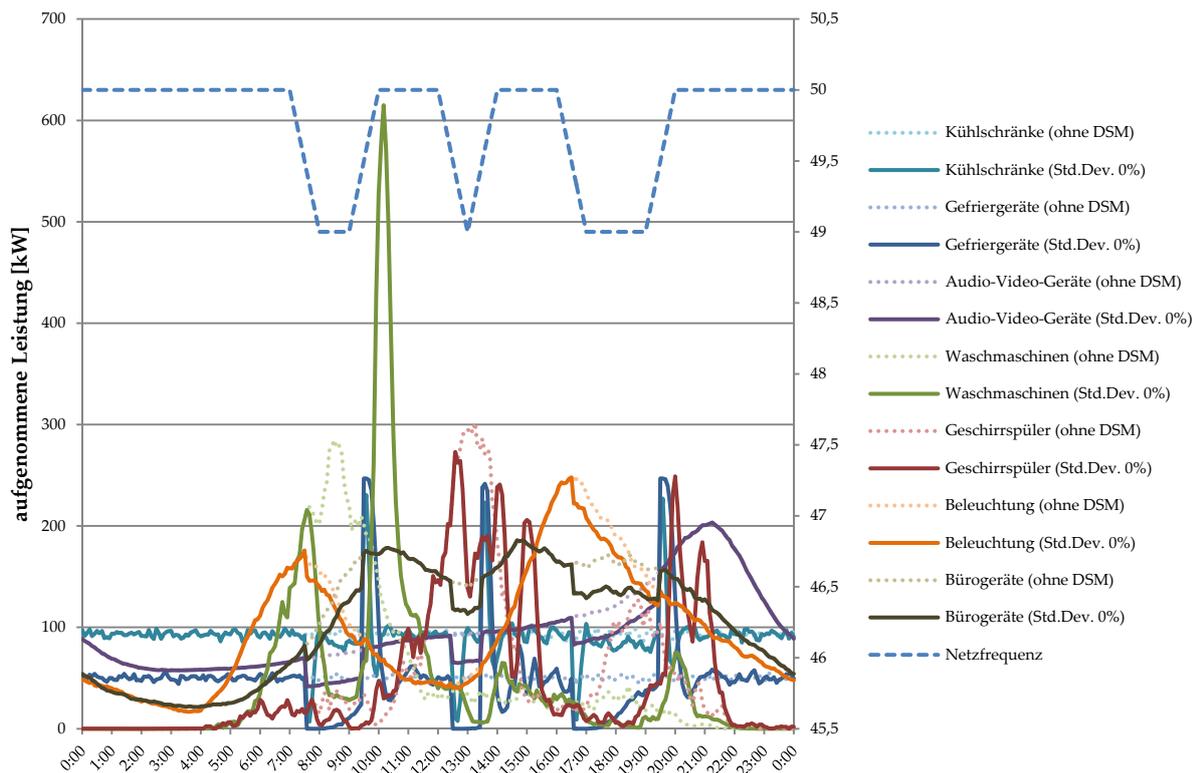


Diagramm 5.2: Simulationsergebnis für das angegebene Szenario mit einer Standardabweichung von `Frequency_Level` von 0%

In Diagramm 5.2 ist ein Sonderfall dargestellt. Hier wurde angenommen, dass die Reaktionsschwelle für alle DSM-Einheiten exakt gleich ist. Zwar kommt es beim Frequenzeinbruch zu einer direkten Leistungsbeeinflussung, bei der Erholung der Netzfrequenz treten aber deutliche Lastspitzen auf, insbesondere bei Kühlgeräten, Waschmaschinen (um ca. 10:00 Uhr) und Geschirrspülern (um ca. 20:00 Uhr).

Gut ist zu erkennen, wie die Gerätegruppen reagieren, die eine Lastreduktion durchführen (Audio-Video-Geräte, Beleuchtung und Bürogeräte). Es kommt zu einer sofortigen Leistungsabnahme um den

angestrebten Wert. Bei einer Erholung der Netzfrequenz kehren diese Geräte wieder in ihre normale Betriebsweise zurück, verursachen jedoch dabei keine erhöhte Leistungsaufnahme.

Überraschend ist bei diesem Szenario, dass auch die Einführung einer Streuung von `Frequency_Level` zu keiner deutlichen Verbesserung führt, wie dies in Diagramm 5.3 und Diagramm 5.4 zu erkennen ist.

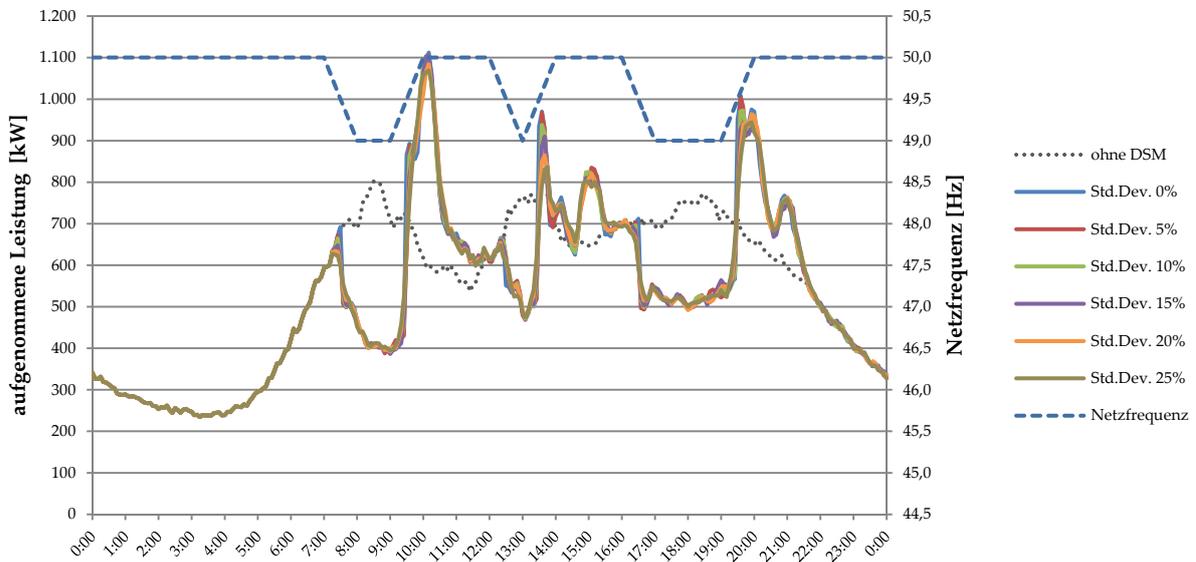


Diagramm 5.3: Auswirkung verschiedener Standardabweichungen des Parameters `Frequency_Level` auf die Gesamtleistung

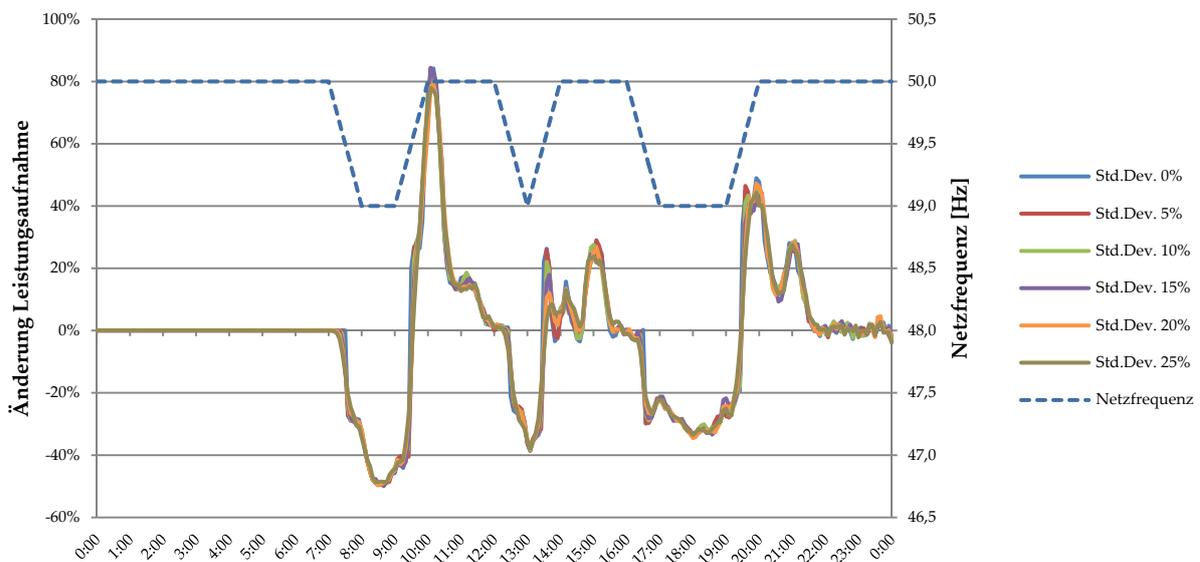


Diagramm 5.4: Relative Änderung der Gesamtleistungsaufnahme bezogen auf die Leistungsaufnahme ohne Einsatz von DSM für unterschiedliche Standardabweichungen des Parameters `Frequency_Level`

Trotz der langsamen Änderung der Frequenz führt eine Erhöhung der Standardabweichung von `Frequency_Level` nur dazu, dass die Verläufe etwas verschliffen, die Höhe der Lastspitzen aber kaum beeinflusst werden. Der Unterschied zwischen 0% und 25% beträgt bei der höchsten Lastspitze (um 10:10 Uhr) gerade einmal 42,8 kW bzw. 3,8%.

Dieses Ergebnis lässt darauf schließen, dass die Frequenzmessung einer DSM-Einheit nicht sehr genau erfolgen muss, um ähnliche Resultate zu erzielen. Sie kann damit auch dementsprechend einfach und damit kostengünstig implementiert werden. Auch kann auf den zusätzlichen Aufwand einer absichtlichen Streuung der Frequenzlevel bei der Produktion verzichtet werden, da die zu erzielenden Effekte, zumindest in dem hier betrachteten Szenario, sehr klein ausfallen.

5.1.2. Rollout von DSM

Interessant ist auch, wie sich die Leistungsreduzierung verhält, wenn nach und nach immer mehr Geräte mit DSM-Einheiten ausgestattet werden, da nie davon ausgegangen werden kann, dass schlagartig alle Verbraucher zur Leistungsreduzierung beitragen.

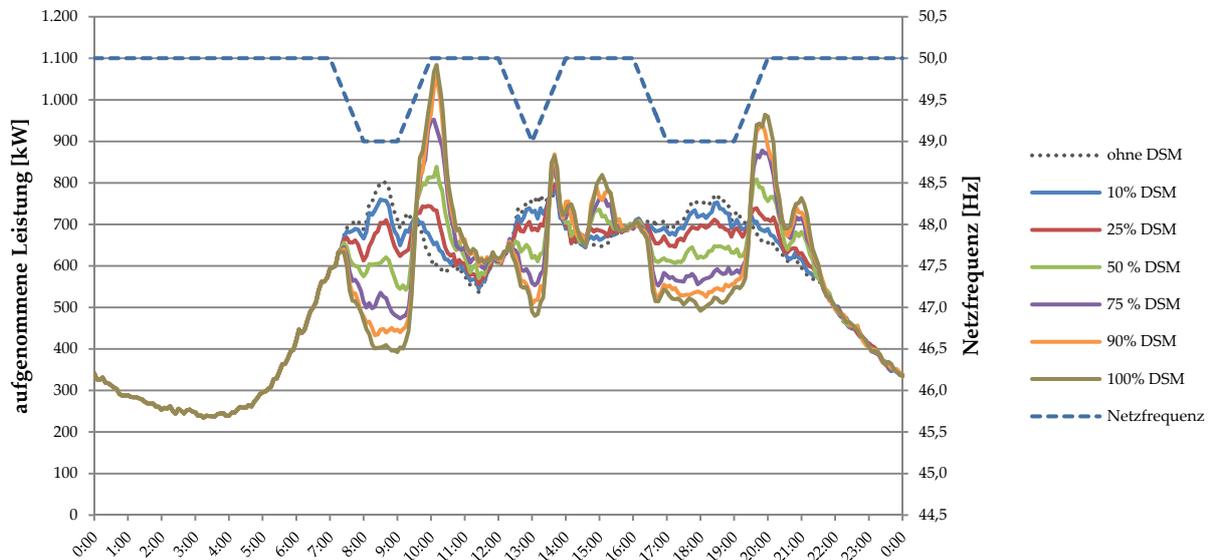


Diagramm 5.5: Auswirkung unterschiedlicher Anteile von Geräten mit DSM auf die Gesamtleistung

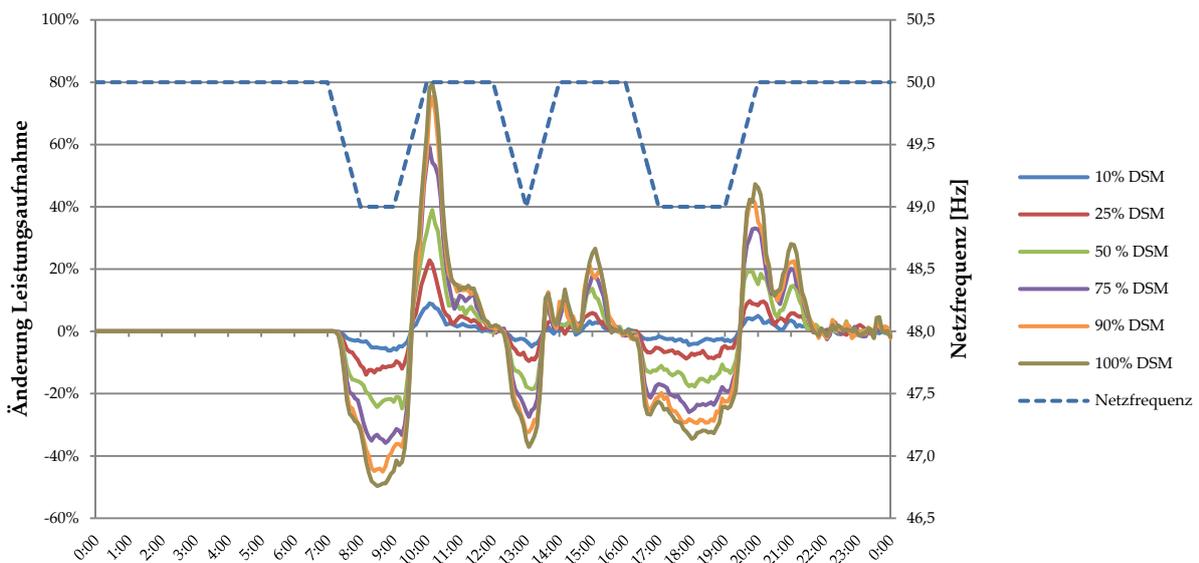


Diagramm 5.6: Änderung der Gesamtleistungsaufnahme bezogen auf die Leistungsaufnahme ohne Einsatz von DSM für unterschiedlichen Anteil von Geräten mit DSM

Diagramm 5.5 und Diagramm 5.6 zeigen so eine Entwicklung für die bisher verwendeten Geräteinstanzen. Als Standardabweichung der Parameters `Frequency_Level` wurde hier ein Wert von 20% verwendet. Weiters wurde bei diesen Durchläufen kein Gerätetyp bei der Einführung von DSM bevorzugt bzw. vernachlässigt.

Es ist erkennbar, dass mit immer höherem Anteil von Geräten mit DSM auch dementsprechend der Umfang der Leistungsreduktion steigt. Bereits ab einem Anteil von 25% sind bereits Reduktionen der Leistungsaufnahme von fast 10% zu erreichen.

5.1.3. Verhalten von Waschmaschinen

Wie man in Diagramm 5.2 erkennen kann, tragen insbesondere die Waschmaschinen bei der ersten Leistungsreduktion zu einer Lastspitze bei. Dies liegt daran, dass alle Waschmaschinen in dem Zeitraum, in

dem die Leistungsreduktion durchgeführt werden soll, ihre Startzeiten verschieben. Dieser Zeitraum ist aber kürzer als zwei Stunden, weshalb nur wenige Waschmaschinen bereits in der Zeit niedriger Frequenz ihr Programm starten. Hat sich die Frequenz erholt, beginnen nun alle Waschmaschinen gleichzeitig mit ihrem Programm, was zu der beobachteten, stark erhöhten Leistungsaufnahme führt.

Nun werden speziell für Waschmaschinen verschiedene Einstellungsmöglichkeiten für die DSM-Output-Moden untersucht, um ein besseres Verhalten zu erhalten.

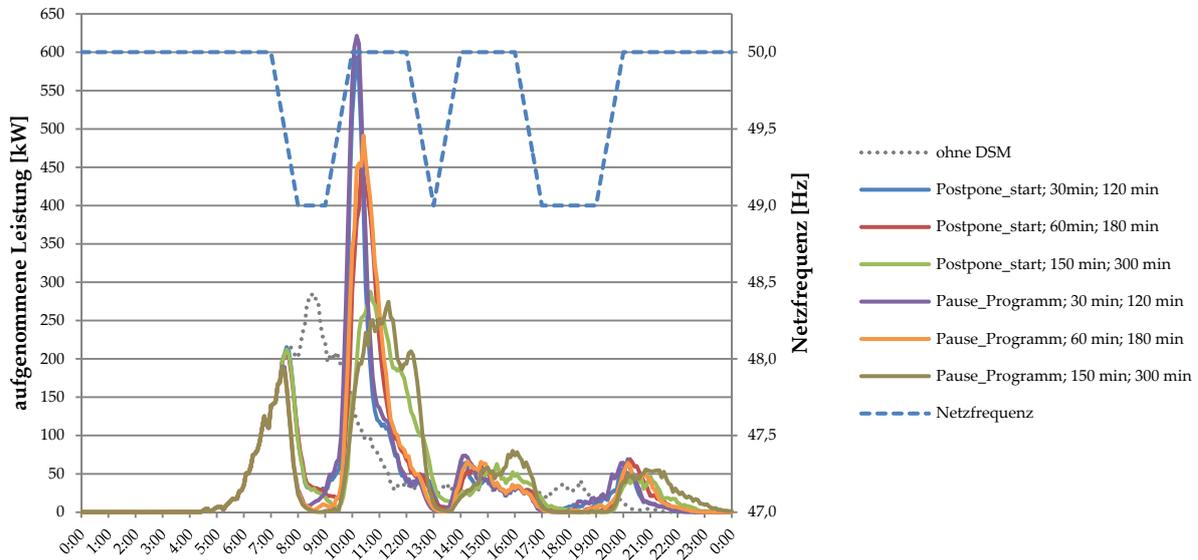


Diagramm 5.7: Auswirkung unterschiedlicher DSM-Output-Moden und Einstellungen auf die Leistungsaufnahme der Waschmaschinen. Leseweise der Legende: DSM-Output-Mode; Time_Postpone_Start; Time_Postpone_max

In Diagramm 5.7 werden die beiden DSM-Output-Moden `Postpone_Start` und `Pause_Programm` mit unterschiedlichen Parametern verglichen. Deutlich ist zu erkennen, dass es bei `Pause_Programm` schneller zu einer Leistungsreduktion kommt, als bei `Postpone_Start`, da die laufenden Programme unterbrochen werden.

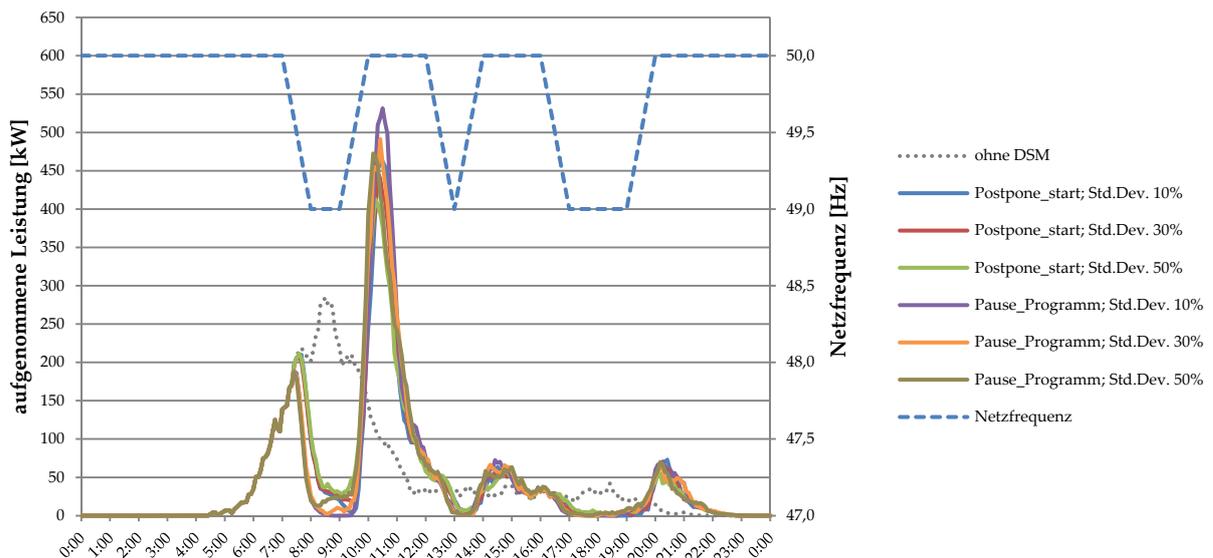


Diagramm 5.8: Auswirkung unterschiedlicher Standardabweichungen der Parameter `Time_Postpone_Start` und `Time_Postpone_max` auf die Beeinflussung der Leistungsaufnahme von Waschmaschinen

Der wichtigste Parameter, mit dem die Leistungsspitze bei den Waschmaschinen reduziert werden kann, ist in diesem Fall `Time_Postpone_Start`. Ist dieser größer als die Zeit, in der die Netzfrequenz reduziert ist, wird der normale Lastverlauf einfach zu jenem Zeitpunkt hin verschoben, in dem die Netzfrequenz wieder einen normalen Wert erreicht (hell- und dunkelgrüne Kurve). Die maximale Verschiebungszeit wurde mit sechs Stunden sehr hoch angesetzt und hat hier keinen Einfluss. Würde sie entspre-

chend reduziert werden, würden die Geräte unabhängig vom Netzzustand ihr Programm starten (siehe Abschnitt 2.3.2.5).

Wie sich die Standardabweichungen der Parameter `Time_Postpone_Start` und `Time_Postpone_max` auf die Leistungsaufnahme auswirken, zeigt Diagramm 5.8. Die Werte sind hier für `Time_Postpone_Start` = 60 min und für `Time_Postpone_max` = 180 min. Deutlich ist auch hier zu erkennen, dass der Einfluss der Streuung eher gering ist und kaum zum gewünschten Effekt einer verminderten Lastspitze bei der Erholung der Netzfrequenz führt.

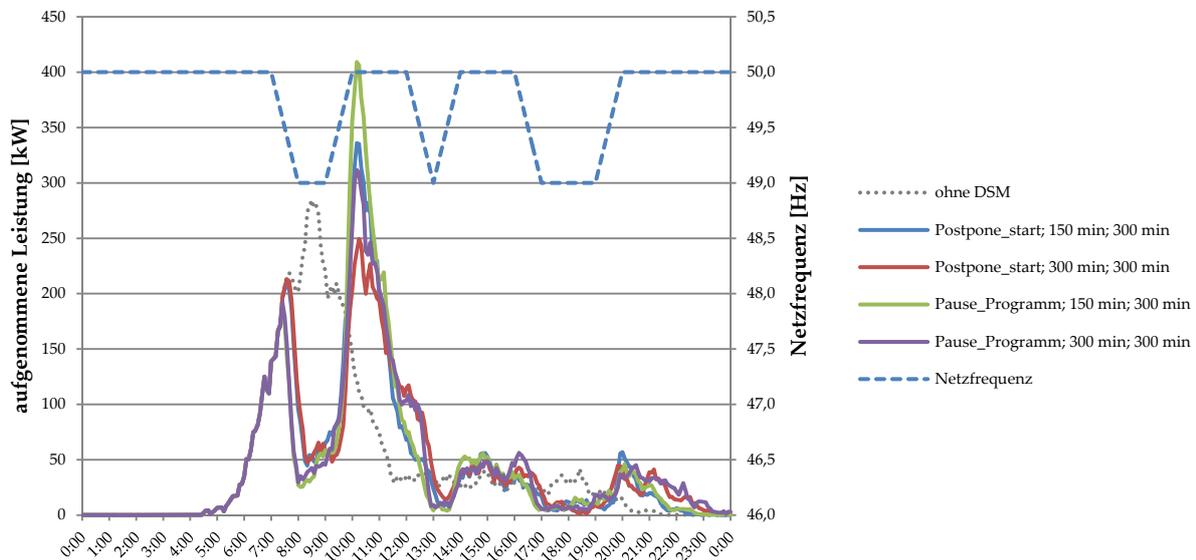


Diagramm 5.9: Gleichverteilung der Parameter `Time_Postpone_Start` und `Time_Postpone_max` bei Waschmaschinen. Leseweise der Legende: DSM-Output-Modus; `Time_Postpone_Start`; `Time_Postpone_max`

Anders sieht es aus, wenn die beiden Parameter `Time_Postpone_Start` und `Time_Postpone_max` nicht gemäß einer Standardabweichung sondern gemäß einer Gleichverteilung gestreut werden, dargestellt in Diagramm 5.9.

Da es durch die Gleichverteilung sämtliche Werte der maximalen Verschiebungszeit des Programmstarts `Time_Postpone_max` gleich wahrscheinlich gibt (siehe Abschnitt 2.1.2), führt dies dazu, dass bereits einige Geräte vor der Erholung der Netzfrequenz ihr Programm starten (sehr gut erkennbar anhand der violetten Kurve).

Bei der Streuung der Parameter gemäß einer Standardverteilung werden auch Werte ermittelt, die größer als der Mittelwert sind. Bei der vorliegenden Implementierung der Gleichverteilung treten jedoch nur Werte auf, die kleiner als der angegebene Mittelwert sind. Um hier eine bessere Vergleichbarkeit herzustellen, müssen die angegebenen Parameterwerte dementsprechend höher angenommen werden, was bei der Erstellung von Diagramm 5.9 berücksichtigt wurde.

Hier zeigt sich, dass das Unterbrechen laufender Programme bei der Erholung der Netzfrequenz zu einem deutlich schlechteren Ergebnis führt als jenes bei einer Standardverteilung, bei dem die Leistungsspitzen bei gleicher Parameterwahl vom Output-Modus unabhängig sind. Eine Erhöhung des Wertes für `Time_Postpone_Start` könnte hier das Ergebnis verbessern, nur scheinen Startzeitverzögerungen von bis zu fünf Stunden bereits als starker Eingriff in das Geräteverhalten, weshalb hier von einer weiteren Erhöhung abgesehen werden muss.

Die bisher betrachteten Output-Moden setzen eine hohe Intelligenz der Geräte voraus, die für das DSM-System verwendet werden können, da die zeitliche Steuerung der Startzeitverschiebung von der Programmsteuerung übernommen werden muss. Für die DSM-Einheit wurde aber auch die Möglichkeit geschaffen, Verzögerungen bei der Wiederinbetriebsetzung der Geräte zu erzielen, in dem die Meldung über einen ungünstigen Netzzustand länger, als dieser real vorherrscht, dem Gerät mitgeteilt wird (siehe Abschnitt 2.3.1.2.2, Parameter `Time_Delay_Restore_Op`).

Im folgenden Beispiel wurde angenommen, dass die Waschmaschinen nur solange ihren Betrieb unterbrechen, solange der DSM-Output anliegt. Zusätzlich wurde eine mittlere maximale Zeit für eine Unterbrechung des Betriebs von zwei Stunden mit einer Standardabweichung von 30% (entspricht 36 min) angenommen, um die Dauer der Verzögerung der normalen Funktion einzuschränken und für den Benutzer kalkulierbarer zu machen.

Diese Beschränkung der maximalen Zeit für eine Funktionsunterbrechung bzw. Startzeitverschiebung könnte auch durch die DSM-Einheit realisiert werden, wodurch keine hohen Anforderungen mehr an die Programmsteuerung des Gerätes gestellt werden müssen.

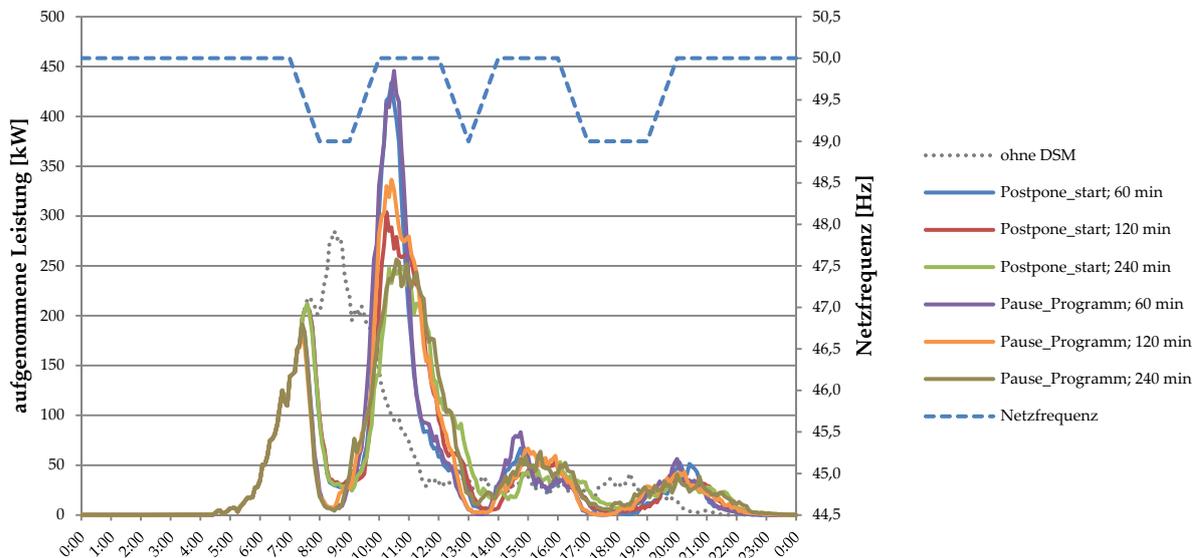


Diagramm 5.10: Auswirkung unterschiedlicher Werte für den Parameter `Time_Delay_Restore_Op` bei Waschmaschinen. Leseweise der Legende: DSM-Output-Mode; `Time_Delay_Restore_Op`

Diagramm 5.10 zeigt Ergebnisse für verschiedene Werte von `Time_Delay_Restore_Op`, die gemäß einer Gleichverteilung gestreut wurden. Das beste Ergebnis für das hier betrachtete Szenario kommt bei sehr hohen Werten zustande (240 min) wobei das Ergebnis kaum vom gewählten Output-Modus abhängt. Hier bietet `Pause_Programm` den Vorteil, schneller auf einen Frequenzeinbruch zu reagieren. Die Wahl hoher Werte für `Time_Delay_Restore_Op` ist aber in dem hier betrachteten Beispiel nicht von großer Relevanz, da jede einzelne Waschmaschine im Mittel nur um 120 min ihren Betrieb unterbricht.

Wegen dieser guten Einschätzbarkeit des Einflusses der Leistungsreduktion und der geringen Anforderungen an das Steuerungsprogramm der Waschmaschine bietet dieser Modus klare Vorteile für eine mögliche Umsetzung in einem DSM-System.

5.1.4. Verhalten von Geschirrspülern

Ähnlich den Waschmaschinen haben auch die Geschirrspüler das Potential durch ungünstige Einstellungen des DSM-Systems unerwünschte Leistungsspitzen zu erzeugen. Im vorliegenden Szenario (vergleiche Diagramm 5.2 bzw. blaue Kurve in Diagramm 5.9) fallen diese aber anscheinend nicht so ins Gewicht.

Hier macht sich der Umstand bemerkbar, dass der gewählte Frequenzverlauf zur Reduzierung der Lastspitze um 13:00 Uhr etwas ungünstig liegt, da bereits viele Geräte zum Zeitpunkt des Einsetzens der Leistungsreduktion aktiv sind, damit ihr Programm gestartet haben und beim DSM-Output-Modus `Postpone_Start` diese laufenden Programme noch beendet werden (blaue und rote Kurve in Diagramm 5.11). Daher fällt einerseits die erzielte Leistungsreduktion, andererseits auch die Leistungsspitze nach der Erholung der Netzfrequenz dementsprechend geringer aus.

Im Output-Modus `Pause_Programm` kann wiederum ein deutlich höheres Reduzierungspotential erreicht werden, das sogar ein fast vollständiges Abschalten der Geschirrspüler um 13:00 Uhr bewirkt (grüne und violette Kurve). Dies führt dann aber bei einer ungünstigen Wahl der Parameter zu einer stark erhöhten Leistungsspitze bei der Erholung der Netzfrequenz. Deutlich sind in diesen Fällen zwei

aufeinander folgende Lastspitzen zu erkennen, die sich aufgrund der gewählten mittleren Lastkurve und dem gleichzeitigen Fortsetzen und Starten von Programmen ergeben.

Durch eine ähnliche Parameterwahl wie bei den Waschmaschinen kann aber auch hier die Situation verbessert werden (violette Kurve).

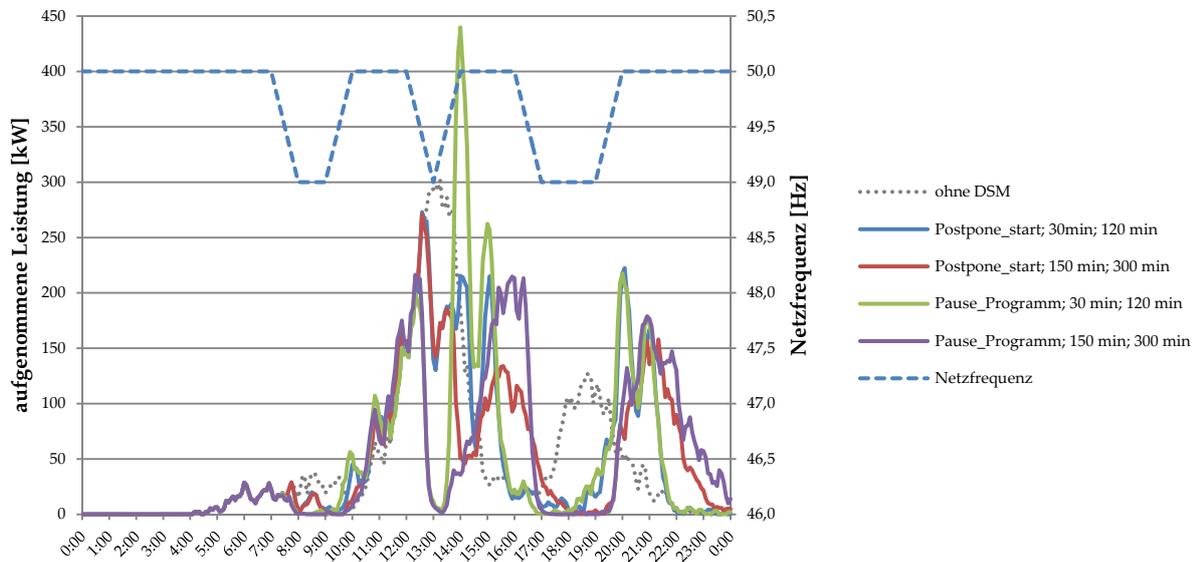


Diagramm 5.11: Auswirkung unterschiedlicher DSM-Output-Moden und Einstellungen auf die Leistungsaufnahme von Geschirrspülern, Standardabweichung der Parameter 30%

Leseweise der Legende: DSM-Output-Mode; Time_Postpone_Start; Time_Postpone_max

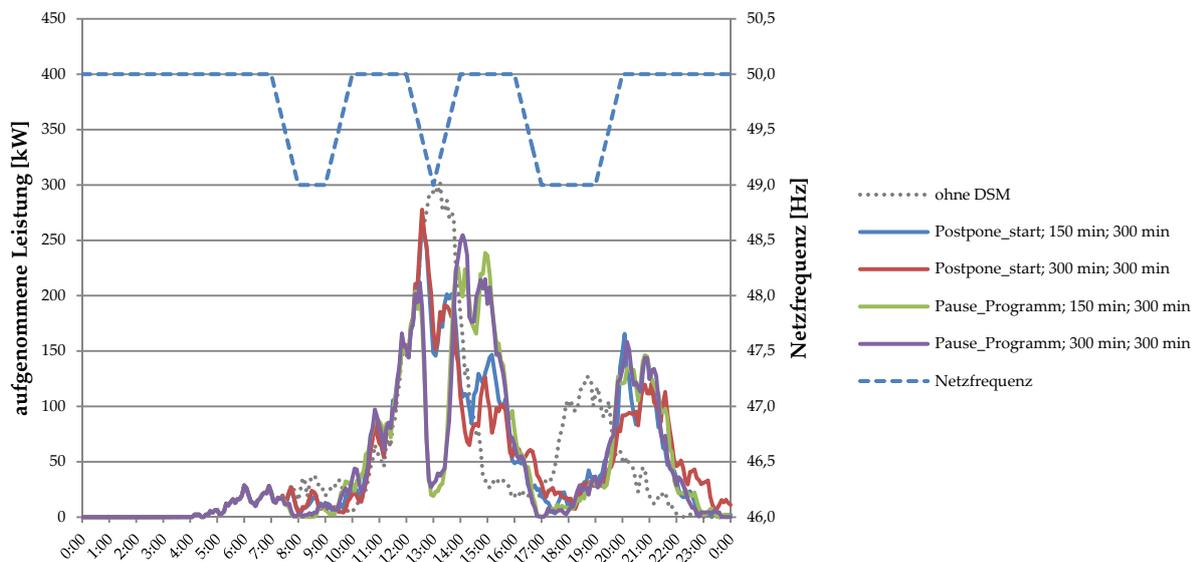


Diagramm 5.12: Gleichverteilung der Parameter Time_Postpone_Start und Time_Postpone_max bei Geschirrspülern. Leseweise der Legende: DSM-Output-Mode; Time_Postpone_Start; Time_Postpone_max

Diagramm 5.12 zeigt die Auswirkung bei einer Gleichverteilung der gestreuten Parameterwerte. Hier ergibt sich ein ähnliches Bild wie bei den Waschmaschinen: Das Potential der Leistungsreduktion fällt genauso wie die Höhe der Leistungsspitzen bei der Erholung der Netzfrequenz niedriger aus.

Wie bei den Waschmaschinen kann auch hier untersucht werden, ob eine Übernahme der Zeitverzögerung durch die DSM-Einheit zu einem brauchbaren Ergebnis führt. Es wurden hierbei die gleichen Werte herangezogen, die auch bei den Waschmaschinen verwendet wurden. Dies soll die Möglichkeit verdeutlichen, dass möglichst gleiche Chips in allen Geräten verwendet werden sollen und somit keine Unterscheidung dieser notwendig ist.

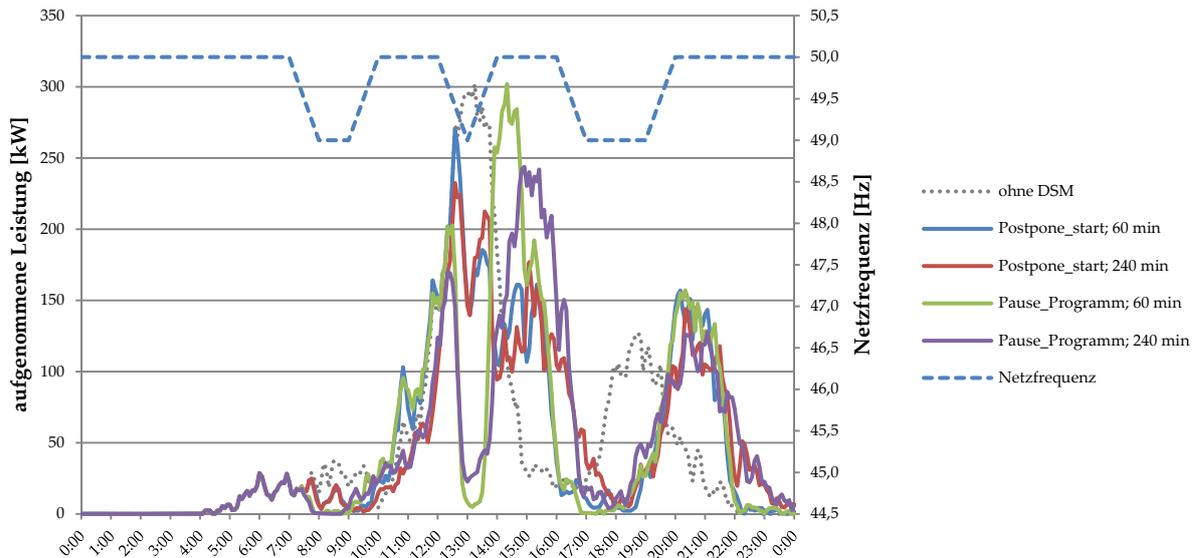


Diagramm 5.13: Auswirkung unterschiedlicher Werte für den Parameter `Time_Delay_Restore_Op` bei Geschirrspülern

Diagramm 5.13 zeigt einige Ergebnisse für diese Annahmen. Auch hier zeigt sich, dass trotz des einfacheren Systems ein für das betrachtete Szenario gutes Ergebnis zustande kommt. Es bietet auch hier der Modus `Pause_Programm` den Vorteil der schnelleren Reaktion auf einen Frequenzeinbruch, führt aber dann dementsprechend zu einer höheren Leistungsspitze nach der Erholung der Netzfrequenz.

Ansonsten kann aber beobachtet werden, dass die erhöhte Leistungsaufnahme bei großen Werten von `Time_Delay_Restore_Op` (240 min) gut aus den Zeiträumen reduzierter Frequenz zu jenen mit normaler Frequenz verschoben werden. Die maximale Leistungsaufnahme fällt jedoch nicht höher, eher sogar geringer aus als beim Betrieb ohne DSM. Trotzdem wird bei jedem Geräte im Mittel der Betrieb nur für 120 Minuten beeinflusst!

5.1.5. Verhalten von Kühlgeräten

Kühlgeräte können auch enorme Lastspitzen verursachen, da sie eine Speichercharakteristik aufweisen. Die zuvor dem Speicher entnommene Energie muss wieder aufgewendet werden, um das Speicherniveau auf die ursprüngliche Höhe zu bringen. Dazu kommen noch die Verluste bei der Energieumwandlung.

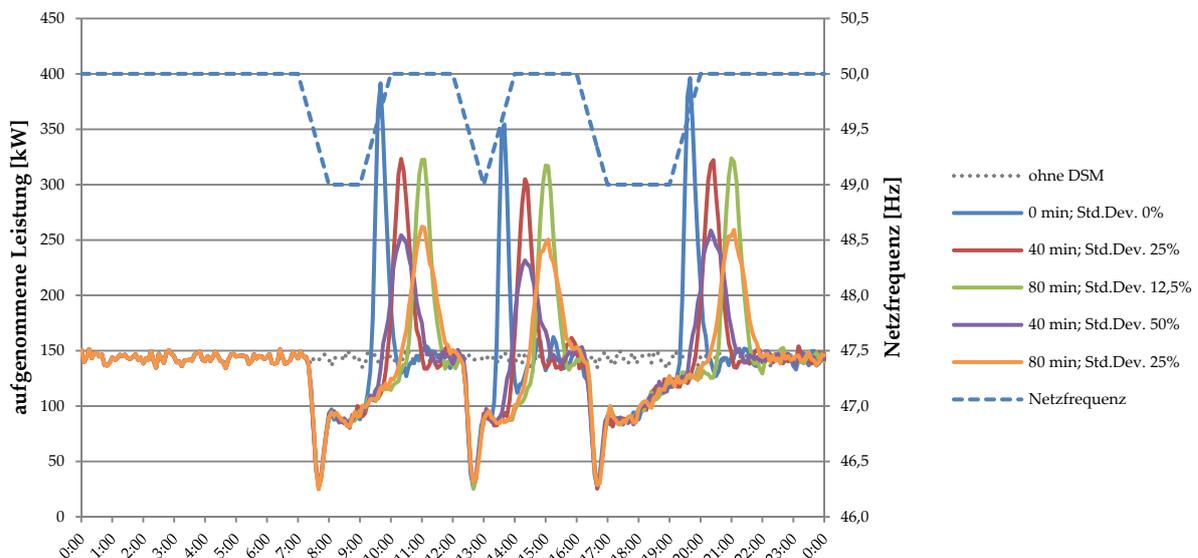


Diagramm 5.14: Einfluss des Parameters `Time_Delay_Restore_Op` mit unterschiedlichen Streuungen gemäß einer Normalverteilung auf die Leistungsaufnahme von Kühlschränken und Gefriergeräten

In Diagramm 5.14 kann dies gut beobachtet werden (blaue Kurve). Um diese Lastspitze zu vermeiden bietet sich bei den Kühlgeräten an, die Rückkehr zur normalen Betriebsweise mit Hilfe des Parameters `Time_Delay_Restore_Op` zu verzögern. In Diagramm 5.14 wurde dies mit verschiedenen Werten und verschiedenen Standardabweichungen versucht, die gemäß einer Standardverteilung gestreut wurden.

Deutlich ist zu erkennen, dass zwar der Zeitpunkt und auch die Höhe der Lastspitze beeinflusst werden kann, aber nicht in einem ausreichenden Umfang.

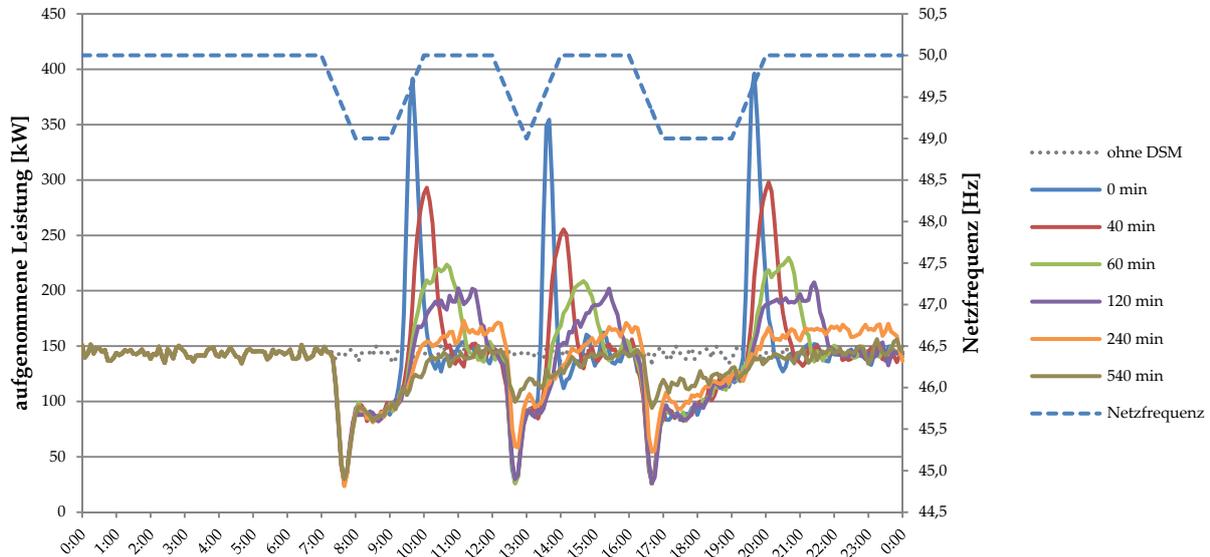


Diagramm 5.15: Einfluss des Parameters `Time_Delay_Restore_Op` mit Streuung gemäß einer Gleichverteilung auf die Leistungsaufnahme von Kühlschränken und Gefriergeräten

Werden hingegen die Werte für `Time_Delay_Restore_Op` gemäß einer Gleichverteilung gestreut, kommt ein deutlich besseres Ergebnis schon bei relativ kleinen Werten für die maximale Verzögerungszeit zustande, dargestellt in Diagramm 5.15.

Gerade bei Kühlschränken wurde keinerlei Einschränkung in Bezug auf die erlaubte Dauer einer Solltemperaturänderung gemacht, weshalb hier im Prinzip eine Leistungsreduktion ohne später auftretende Lastspitzen möglich wäre (dunkelgrüne Kurve). Bei Gefriergeräten ist das nicht ohne weiteres möglich.

Deutlich ist zu erkennen, dass bei hohen Werten der Verzögerung nicht mehr das volle Potential der Leistungsreduktion bei späteren Frequenzeinbrüchen zu Verfügung steht, da noch einige Geräte mit erhöhter Solltemperatur betrieben werden. Die dann zu beobachtende Leistungsreduktion ergibt sich fast ausschließlich aus der reduzierten Leistungsaufnahme durch die höhere Solltemperatur und nicht mehr durch das „Entleeren“ der Speicher (dunkelgrüne Kurve).

Die hier gezeigte Möglichkeit der Leistungsbeeinflussung wird, bis auf die Solltemperaturerhöhung, vollständig auf der DSM-Einheit mit Hilfe einer einfachen Timer-Funktion realisiert. Um eventuell eine zu lange Temperaturerhöhung zu vermeiden, könnte ähnlich wie bei den Waschmaschinen und Geschirrspülern, eine maximale Zeit der Leistungsreduktion eingeführt werden. Wird diese überschritten, kehrt das Gerät in seine normale Betriebsweise zurück. Diese Funktion könnte durch das Gerät selbst oder durch die DSM-Einheit bewerkstelligt werden. In der vorliegenden Arbeit wurde diese Funktion jedoch nicht mehr implementiert, weshalb der Einfluss dieser nicht untersucht werden kann.

5.1.6. Einsatz identer Chips in allen Geräten

In den vorhergehenden Abschnitten konnte gezeigt werden, dass es zufriedenstellende Ergebnisse im betrachteten Szenario gibt, wenn durch die DSM-Einheit die Dauer der Leistungsreduktion verlängert wird und die maximalen Dauer, die ein einzelnes Gerät beeinflusst wird, beschränkt ist.

Was passiert nun, wenn alle Geräte mit dem gleichen Chip ausgestattet werden, dessen Parameter `Time_Delay_Restore_Op` gemäß einer Gleichverteilung gestreut wurde? Dies würde den Vorteil bie-

ten, dass keinerlei Unterscheidung zwischen den einzelnen Chips getroffen werden muss, was zu einer hohen Stückzahl und damit reduzierten Kosten führen könnte.

Die Umsetzung der Leistungsreduktion muss aber von den Geräten ihren Möglichkeiten entsprechend zur Verfügung gestellt werden. Wie bei den Waschmaschinen und den Geschirrspülern bereits erwähnt, würde aber in diesem Fall die Anforderung an diese sogar geringer ausfallen als bei anderen Beeinflussungsmethoden. Im nun betrachteten Beispiel wird davon ausgegangen, dass alle Geräte diese Bedingung erfüllen und mit DSM-Chips ausgestattet sind.

Die Beschränkung der Beeinflussungszeit sollte eigentlich durch die DSM-Einheit erfolgen. In der aktuellen Implementierung wurde diese Funktion noch nicht eingeführt. Grundannahme für die Wahl der Reduzierungspotentiale in Kapitel 4 war aber, dass bei der Beeinflussung der Kühlgeräte sowie der Audio-Video-Geräte, der Beleuchtung und der Bürogeräte keine gravierenden Einschränkungen in der Benutzbarkeit bzw. der Funktion dieser Geräte vorliegen soll. Ist dies gegeben, wovon hier nun ausgegangen wird, kann eine Leistungsreduktion auch über einen längeren Zeitraum aufrecht erhalten werden, weshalb eine Beschränkung der Beeinflussungszeit nicht unbedingt notwendig ist.

Bei den Geräten, die ein Programm abarbeiten ist diese Beschränkung bereits realisiert (`Time_Postpone_max`) und wird dementsprechend auch genutzt, um ein vorhersehbares Geräteverhalten zu erhalten. Dabei wird bei diesen Geräten der einfache DSM-Output-Modus `Postpone_Start` verwendet.

In Diagramm 5.16 sind die Reaktionen der einzelnen Verbrauchergruppen zusammengefasst. Deutlich ist hier zu erkennen, dass alle Geräte ab dem ersten Frequenzeinbruch um 7:00 Uhr durch die lange Zeitverzögerung dauerhaft in ihrem Verhalten beeinflusst werden.

Hier ist es so, dass jene Geräte, die bereits zu Beginn des Frequenzeinbruches aktiv sind, länger ihre Leistung reduzieren als jene, die kurz vor dem Ende des Einbruchs eingeschaltet werden. Durch die Einführung einer maximalen Beeinflussungszeit auch bei den restlichen Gerätegruppen (also Kühlgeräte, Audio-Video-Geräte, Beleuchtung und Bürogeräte) könnte die Leistungsreduktion gleichmäßiger auf alle Geräte aufgeteilt werden.

Der Verlauf der Gesamtleistung kann Diagramm 5.17 entnommen werden.

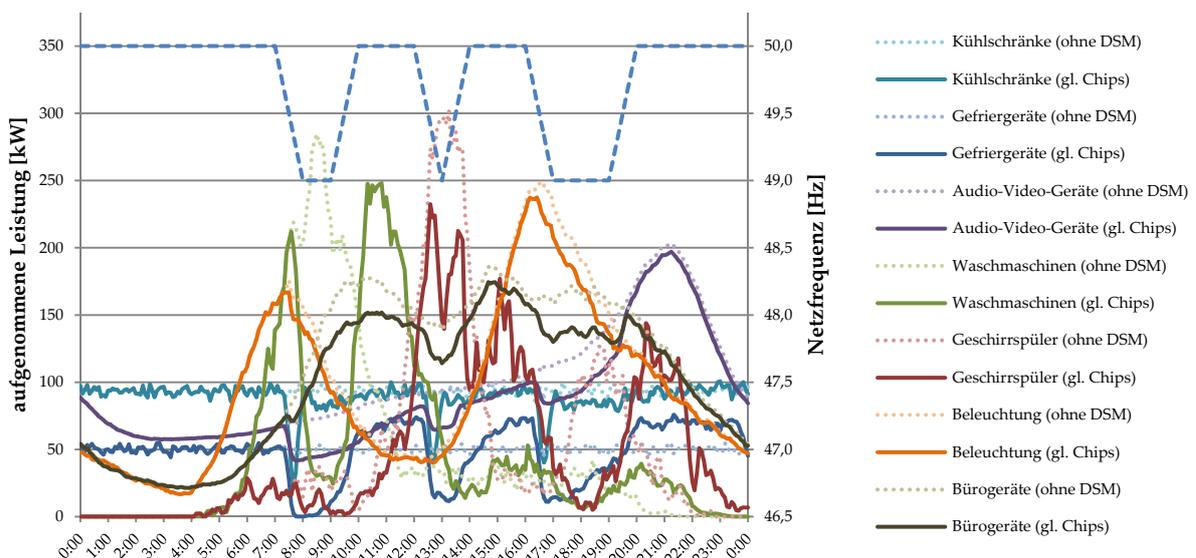


Diagramm 5.16: Simulationsergebnis für das betrachtete Szenario für Einsatz identischer DSM-Chips in allen Geräten, `Time_Delay_Restore_Op` = 240 min gleichverteilt; `Time_Postpone_max` = 120 min mit 30% Standardabweichung;

5.1.7. Zusammenfassung

Das hier betrachtete Szenario, bei dem versucht wird, mit Hilfe der Netzfrequenz die Spitzen der Leistungsaufnahme zu verschieben, stellt natürlich nur eine von sehr vielen anderen Möglichkeiten dar, wie die Netzfrequenz verlaufen könnte.

Auch wurde noch keine genauere Definition bzw. Untersuchung von möglichen DSM-Systemen hier durchgeführt. Vielmehr sollten hier nur erste Beispiele für eine mögliche Umsetzung vorgestellt werden. Dennoch lassen diese Beispiele bereits erste Erkenntnisse zu, trotz des beschränkten Umfangs der hier gemachten Simulationen.

So zeigt sich, dass die Streuung der Frequenzlevel keinen großen Einfluss auf die Gesamtreaktion des Systems hatte. Dies kann dahingehend gedeutet werden, dass eine exakte Messung der Netzfrequenz nicht unbedingt notwendig ist.

Mit Hilfe der zeitlichen Verzögerung der Beendigung der Leistungsbeeinflussung in Verbindung mit einer maximalen Dauer von dieser kann eine drastische Erhöhung der aufgenommenen Leistung vermieden werden, insbesondere bei Waschmaschinen und Geschirrspülern.

Das legt den Schluss nahe, dass ein Einsatz von identischen Chips mit gleichen Parametern durchaus zu guten Ergebnissen führen kann, sofern die Leistungsbeeinflussung vom Benutzer der Geräte nicht als störend empfunden wird. Die für dieses Szenario gemachten Annahmen gehen davon aus. Auch wird angenommen, dass eine mittlere Verschiebung der Programmausführung von ca. zwei Stunden bei Geschirrspülern und Waschmaschinen als nicht zu massiver Eingriff angesehen werden kann.

Parameter	DSM unverzögert			DSM selektiv verzögert			DSM-Einheiten identisch		
	Wert	Std.Abw.	Verteilung	Wert	Std.Abw.	Verteilung	Wert	Std.Abw.	Verteilung
Kühlgeräte									
Time_Delay_Restore_Op	0 min	0 %	Normal	120 min	-	Gleich	240 min	-	Gleich
Waschmaschinen									
Time_Postpone_Start	5 min	30 %	Normal	300 min	-	Gleich	0 min	0 %	Normal
Time_Postpone_max	120 min	30 %	Normal	300 min	-	Gleich	120 min	10 %	Normal
Time_Delay_Restore_Op	0 min	0 %	Normal	0 min	0 %	Normal	240 min	-	Gleich
DSM-Output-Modus	Pause_Programm			Postpone_Start			50% Postpone_Start 50% Pause Programm		
Geschirrspüler									
Time_Postpone_Start	5 min	30 %	Normal	150 min	30 %	Normal	0 min	0 %	Normal
Time_Postpone_max	120 min	30 %	Normal	300 min	30 %	Normal	120 min	10 %	Normal
Time_Delay_Restore_Op	0 min	0 %	Normal	0 min	0 %	Normal	240 min	-	Gleich
DSM-Output-Modus	Pause_Programm			Postpone_Start			50% Postpone_Start 50% Pause Programm		
Restliche Geräte									
Time_Delay_Restore_Op	0 min	0 %	Normal	0 min	0 %	Normal	240 min	-	Gleich

Tabelle 5.1: Gegenüberstellung der wichtigsten Parameter der verschiedenen exemplarischen DSM-Systeme

Im Folgenden sollen nun verschiedene DSM-Systeme miteinander verglichen werden. Dazu werden drei definiert und untersucht.

Das erste System ist jenes, das kaum Verzögerungen in seinem Verhalten aufweist. Sobald sich die Netzfrequenz einem normalen Niveau nähert, kehren alle Geräte wieder zu ihrer normalen Betriebsweise zurück. So bleibt die Dauer der Beeinflussung fast nur auf die Dauer des Frequenzeinbruches beschränkt. Dieses System wird im Weiteren mit „DSM unverzögert“ bezeichnet.

Beim zweiten System beschränkt sich die Verzögerung auf die Kühlgeräte und Geräte mit einem Programmablauf. Das sind jene Gerätetypen, die das Potential haben, zu starken Spitzen der Leistungsaufnahme zu führen. Dieses System wird mit „DSM selektiv verzögert“ bezeichnet.

Für das dritte System wurde angenommen, dass alle DSM-Einheiten mit den gleichen Input-Parametern ausgestattet werden. Weiters verzögern diese die Rückkehr zur normalen Betriebsweise um einen relativ hohen Wert (240 Minuten, Gleichverteilung). Bei den Waschmaschinen und Geschirrspülern wurde angenommen, dass die Hälfte der Geräte die Fähigkeit besitzt, ein laufendes Programm zu unterbrechen (Pause_Programm). Im Folgenden wird mit „DSM-Einheiten identisch“ auf dieses System referenziert.

Tabelle 5.1 stellt die gewählten Parameterwerte für die einzelnen Systeme gegenüber. Sofern nicht anders angegeben, entsprechen die sonstigen Einstellungen den Ausführungen aus Kapitel 4. Für den Schwellwert für die Frequenz `Frequency_Level` wurde ein Mittelwert von 0,5 Hz mit einer Standardabweichung von 20 % vom Mittelwert angenommen. Die Ausstattung mit DSM beträgt bei allen Systemen 100%.

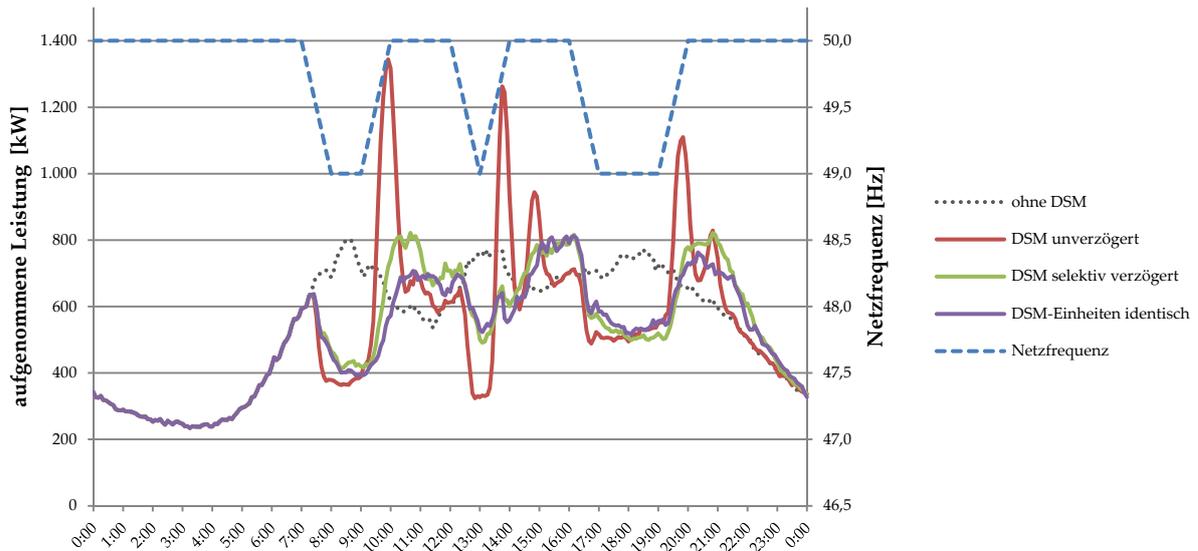


Diagramm 5.17: Vergleich der Gesamtleistungsaufnahme der exemplarischen DSM-Systeme für das Szenario „Verschiebung Lastspitzen“

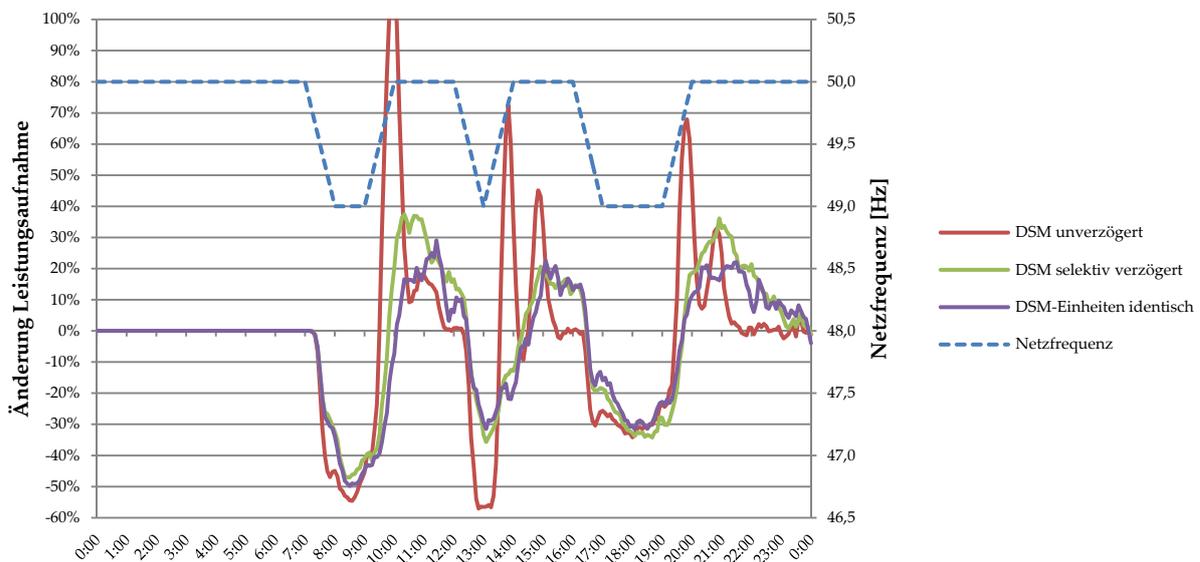


Diagramm 5.18: relative Änderung der Gesamtleistung bezogen auf die Leistungsaufnahme ohne DSM für das Szenario „Verschiebung Lastspitzen“

Mit Hilfe der bisher gemachten Simulationen können nun bereits diese drei DSM-Systeme für das Szenario „Verschiebung Lastspitzen“ verglichen werden. Die Gesamtleistungsaufnahme ist in Diagramm 5.17, die relative Änderung dieser bezogen auf die Leistungsaufnahme ohne DSM in Diagramm 5.18 dargestellt.

Beim unverzögerten System kommt es zu einer deutlich erhöhten Leistungsaufnahme unmittelbar nachdem die Netzfrequenz wieder über den Schwellwert angestiegen ist. Im ungünstigsten Fall wird sogar die Leistungsaufnahme mehr als verdoppelt. Um 14:00 Uhr und um 20:00 Uhr kann man auch deutlich Doppelspitzen erkennen, die sich aufgrund des gleichzeitigen Startens von Geschirrspülern ergeben.

Bei den anderen DSM-Systemen tritt hingegen keine ausgeprägte Lastspitze auf. Es kann aber auch, verglichen mit dem unverzögerten System, hier kurzfristig die Leistung um bis zu 30% reduziert werden, mit Ausnahme der Reduktion um 13:00 Uhr. Hier kann beim unverzögerten System durch den Einsatz des DSM-Output-Modus `Pause_Programm` eine Reduzierung um fast 55% erzielt werden. Die Erhöhung der Leistungsaufnahme bleibt jedoch bei den DSM-Systemen „selektiv verzögert“ und „Einheiten identisch“ stark beschränkt und fällt kaum höher aus als die maximal aufgenommene Leistung der Verbraucher ohne DSM.

Gut ist auch zu erkennen, dass beim Einsatz identischer DSM-Einheiten mit relativ langer Verzögerung bei diesem Szenario ein überraschend gutes Verhalten der Verbraucher erzielt werden kann.

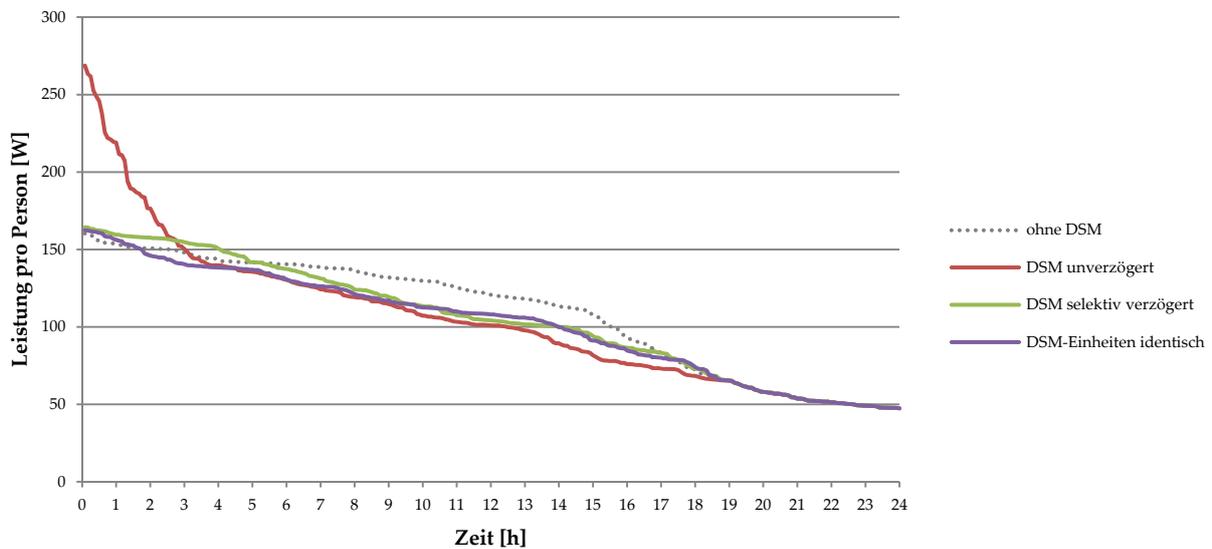


Diagramm 5.19: Lastdauerlinien der Gesamtleistung der verschiedenen DSM-Systeme für das Szenario „Verschiebung Lastspitzen“

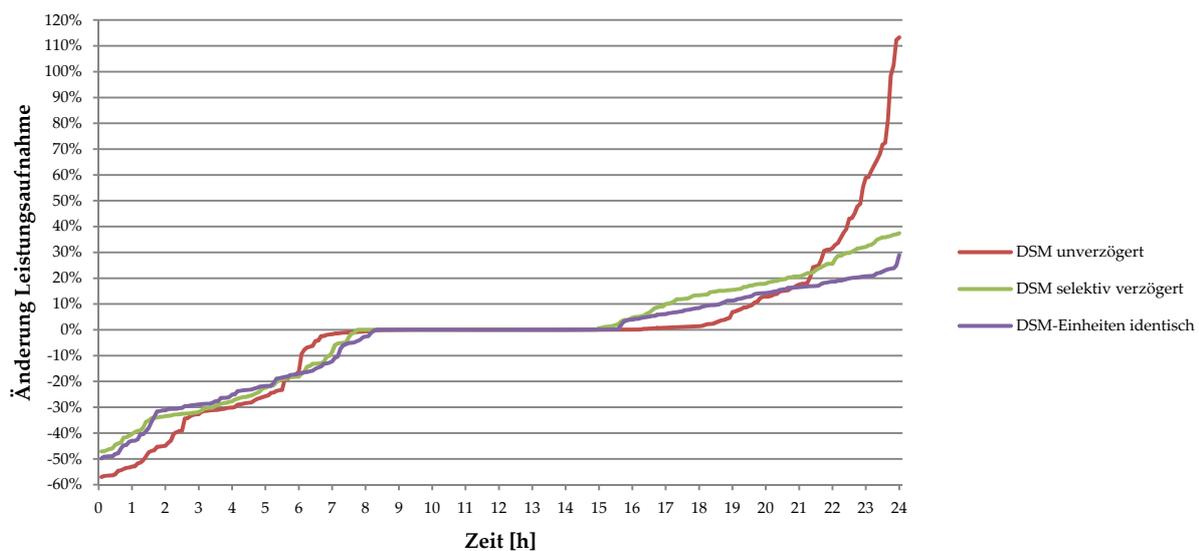


Diagramm 5.20: Dauerlinien der relativen Änderung der Gesamtleistung für das Szenario „Verschiebung Lastspitzen“

In Diagramm 5.19 und Diagramm 5.20 sind die Ergebnisse aus Diagramm 5.17 bzw. Diagramm 5.18 der Größe nach sortiert, wodurch sich Verläufe ergeben, die zeigen, wie lange jeweils die verschiedenen Werte angenommen werden.

Anhand dieser Verläufe kann man gut erkennen, dass zwar das unverzögerte DSM-System eine höhere Reduzierung der Leistung bietet, diese erfolgt aber auf Kosten einer stark erhöhten Leistungsaufnahme zu anderen Zeitpunkten. Im Vergleich dazu sind die beiden DSM-Systeme näher am Verlauf der Leistungsaufnahme ohne DSM, können aber ebenso eine Reduzierung in der gleichen Größenordnung erzielen.

Das DSM-System mit identischen Einheiten schafft es sogar eine Leistungsaufnahme zu erzielen, die, bis auf zwei Stunden, kleiner oder gleich ist als die Gerätekonstellation ohne DSM. Dies begründet sich in der bei Audio-Video-, Bürogeräten und Beleuchtung durchgeführten Leistungsreduktion, die hier für einen längeren Zeitraum erfolgt als bei den anderen DSM-Systemen (siehe Diagramm 5.16).

Bereits anhand dieses einfachen Szenarios zeigt sich, dass relativ schnell Anforderungen und mögliche günstige Einstellungen für ein DSM-System gefunden werden können, welches nur auf die Netzfrequenz reagiert. Inwieweit diese auch für andere Frequenzverläufe gültig sind, soll anhand anderer Szenarien untersucht werden.

5.2. Reaktion auf kurzen Frequenzeinbruch

Beim bisherigen Szenario hat sich die Netzfrequenz relativ langsam geändert und hat ihren Wert für lange Zeiträume geändert.

Nun soll untersucht werden, wie die DSM-Systeme, die im vorhergehenden Abschnitt definiert wurden, auf kurze Einbrüche der Netzfrequenz reagieren. Dazu wurde angenommen, dass sich die Frequenz zu vier Zeitpunkten innerhalb von 5 Minuten um 1 Hz reduziert, 5 Minuten diesen Wert hält dann schließlich in 5 Minuten wieder auf den Ausgangswert zurückkehrt. Diese Einbrüche finden jeweils um 3:00, 8:00, 13:00 und 18:00 Uhr statt.

Der Verlauf der Netzfrequenz kann Diagramm 5.21 entnommen werden.

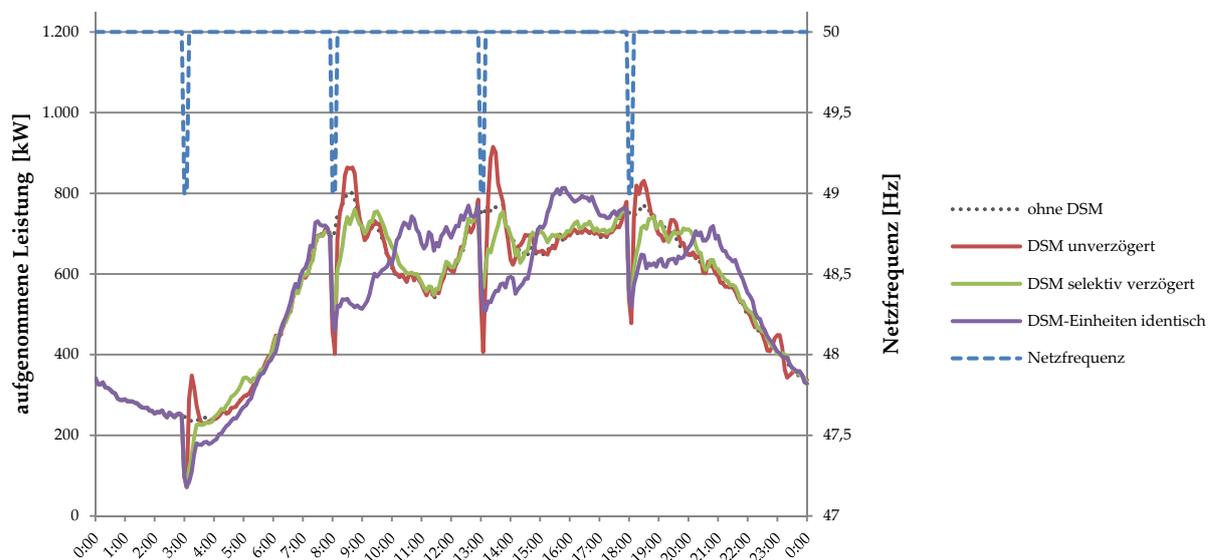


Diagramm 5.21: Vergleich der Gesamtleistungsaufnahme für die verschiedenen DSM-Systeme für das Szenario „kurzer Frequenzeinbruch“

Um 3:00 Uhr kann, relativ gesehen, die größte Reduktion erzielt werden, da zu dieser Zeit hauptsächlich Kühlgeräte in Betrieb sind. Diese stellen kurz ihren Betrieb ein, solange, bis die neue, höhere Solltemperatur erreicht wird.

Bis auf das DSM-System, in dem alle Einheiten identisch sind, kehrt die Leistungsaufnahme bereits bald, nachdem der Frequenzeinbruch vorüber ist, zum normalen Niveau zurück.

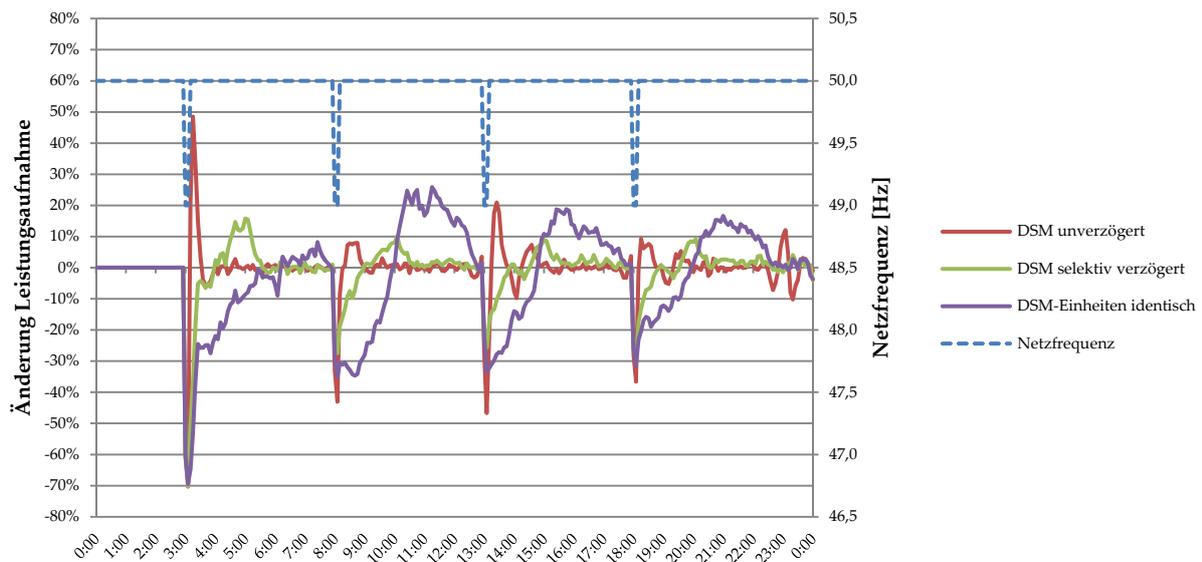


Diagramm 5.22: relative Änderung der Gesamtleistung bezogen auf die Leistungsaufnahme ohne DSM für das Szenario „kurzer Frequenzeinbruch“

Durch die lange Verzögerung beim DSM-System „Einheiten identisch“ kommt es zu einem deutlich anderen Ergebnis. Die kurzen Frequenzeinbrüche reichen aus, um eine Verschiebung der Lastspitzen hin zu späteren Zeitpunkten zu bewirken. Dies zeigt das Potential dieses Systems, durch kurze, eventuell gesteuerte Eingriffe in die Netzfrequenz bereits deutliche Auswirkungen auf das Lastverhalten zu erreichen. So kann mit den getroffenen Einstellungen die Leistung für die nächsten 90 Minuten nach dem Frequenzeinbruch um fast 20% reduziert werden, danach wird keine höhere Zunahme als ca. 20% erreicht.

Sind die Frequenzeinbrüche nicht beabsichtigt, werden hier aber viele Geräte länger in ihrer Leistung beeinflusst, als dies eigentlich notwendig wäre. Im ungünstigsten Fall muss aufgrund eines Frequenzeinbruches in der Dauer von 15 min ein Gerät vier Stunden eine Leistungsreduktion durchführen. Führt dies zu einer Beeinträchtigung der Nutzung, muss dieses Verhalten als nicht optimal betrachtet werden.

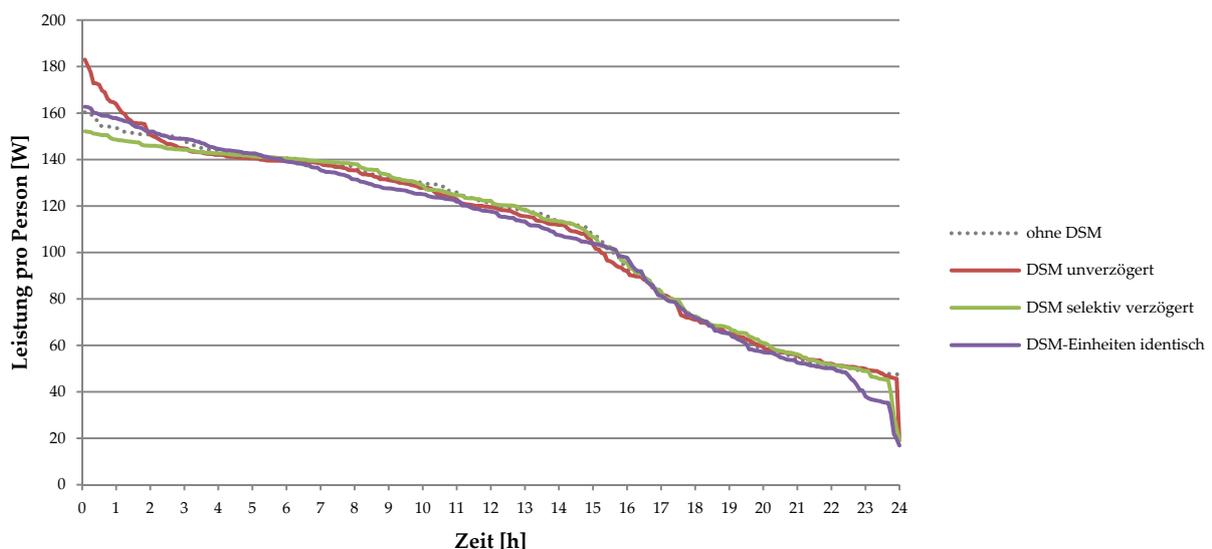


Diagramm 5.23: Lastdauerlinien der Gesamtleistung für das Szenario „kurzer Frequenzeinbruch“

Anhand der Lastdauerlinie in Diagramm 5.23 kann man erkennen, dass alle drei Systeme in Bezug auf ihre Leistungsaufnahme ähnlich dem Gerätekollektiv ohne DSM sind. Die beste Übereinstimmung liefert das System „DSM selektiv verzögert“.

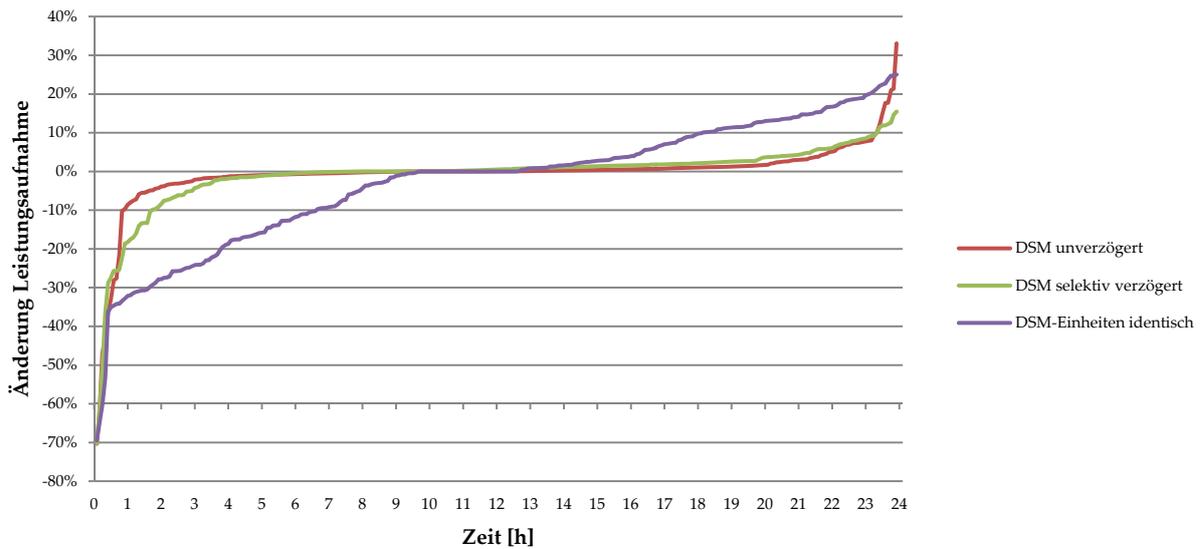


Diagramm 5.24: Dauerlinien der relativen Änderung der Gesamtleistung für das Szenario „kurzer Frequenzeinbruch“

An der Dauerlinie der relativen Änderung der Gesamtleistung kann man auch erkennen, dass das DSM-System mit identischen Einheiten die Leistung über einen längeren Zeitraum beeinflusst als die beiden anderen Systeme. Die kurzfristige hohe Reduktion aller drei erklärt sich hier wiederum aus dem Frequenzeinbruch um 3:00 Uhr, da hier fast nur Kühlgeräte zur Leistungsreduktion beitragen und dies bei allen drei DSM-Systemen gleichartig erfolgt.

5.3. Reaktion auf langen Frequenzeinbruch

Neben den Auswirkungen von kurzen Frequenzeinbrüchen sollen nun die Auswirkungen einer langen Störung der Netzfrequenz untersucht werden. Ob dieser Einbruch aufgrund einer kritischen Versorgungssituation oder aus Steuerungsgründen wegen eines unzureichenden prognostizierten Dargebots erfolgt, ist an dieser Stelle nicht von vordergründigem Interesse. In beiden Fällen sollte jedoch über den gesamten Zeitraum des Frequenzeinbruchs die Leistungsaufnahme reduziert werden.

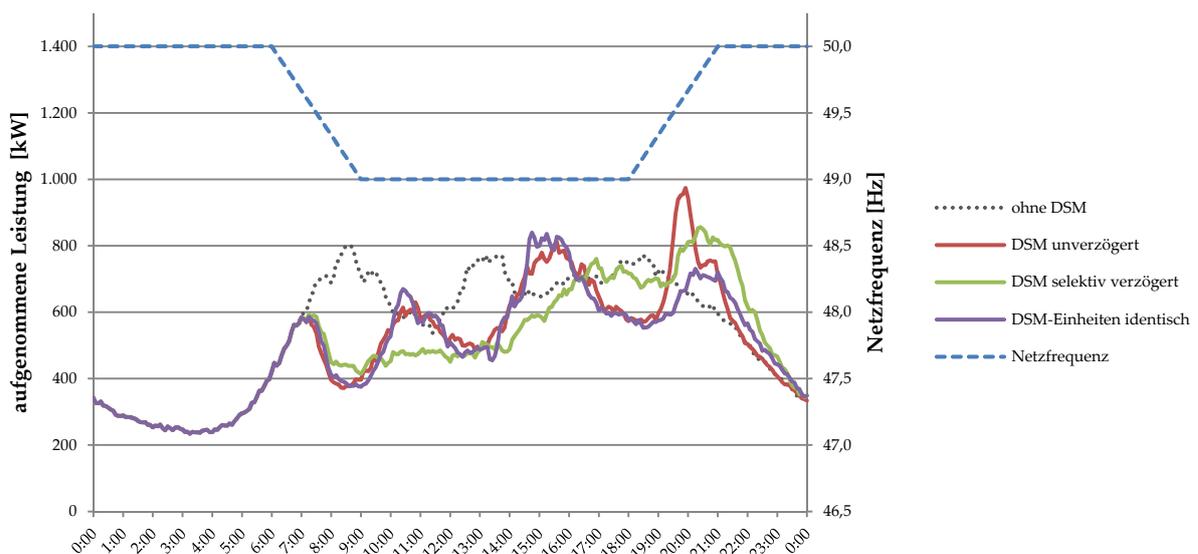


Diagramm 5.25: Vergleich der Gesamtleistungsaufnahme für die verschiedenen DSM-Systeme für das Szenario „langer Frequenzeinbruch“

Für dieses Szenario wurde angenommen, dass die Netzfrequenz von 6:00 bis 9:00 Uhr um 1 Hz abnimmt, dann für neun Stunden konstant bleibt und sich dann zwischen 18:00 und 21:00 Uhr wieder auf das normale Niveau erholt. Der Frequenzverlauf kann Diagramm 5.25 entnommen werden.

Bei den beiden DSM-Systemen „unverzögert“ und „Einheiten identisch“ zeigt sich, dass die Lastspitzen zwar einige Zeit aufgeschoben werden, aber trotzdem auftreten. Dies liegt in der maximalen Verschiebungszeit des Programmes bei Waschmaschinen und Geschirrspülern. Interessant ist dabei, dass beide Systeme ein sehr ähnliches Verhalten aufweisen. Das liegt daran, dass in beiden Fällen die maximale Verschiebungszeit $Time_Postpone_max$ auf 120 Minuten festgelegt wurde (siehe Tabelle 5.1). Die unterschiedliche Standardabweichung fällt dabei nicht ins Gewicht, wie bereits in Abschnitt 5.1.3 festgestellt wurde.

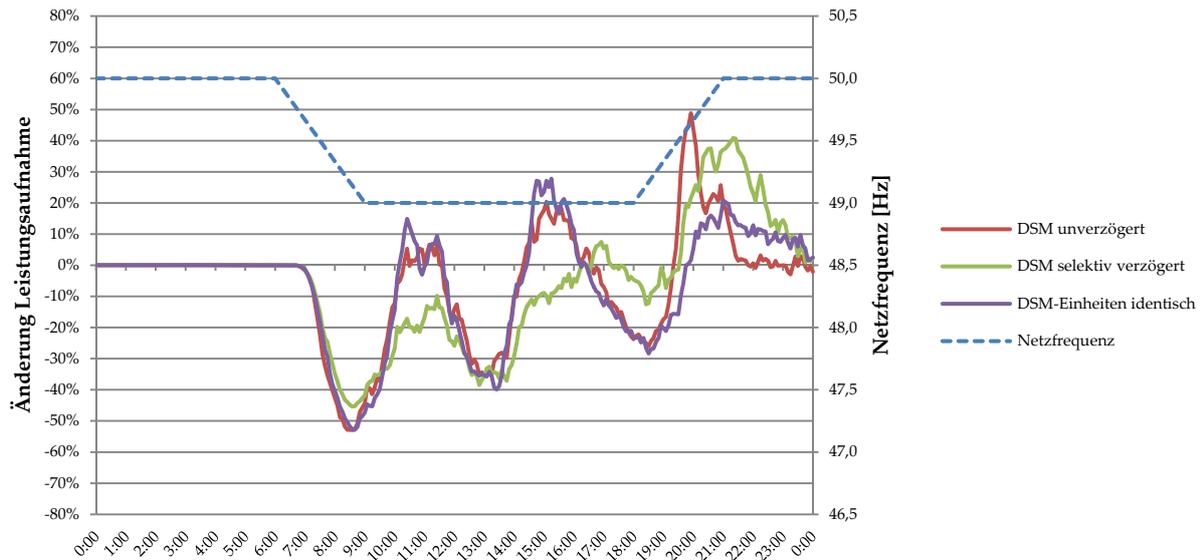


Diagramm 5.26: relative Änderung der Gesamtleistung bezogen auf die Leistungsaufnahme ohne DSM für das Szenario „langer Frequenzeinbruch“

Betrachtet man die relative Änderung der Gesamtleistung, dargestellt in Diagramm 5.26, kann man erkennen, dass diese sich bei allen Systemen lange im negativen Bereich bewegt. Am besten schneidet hier das System „selektiv verzögert“ ab, da aufgrund der sehr langen maximalen Verzögerungszeiten (fünf Stunden) der Betrieb von Waschmaschinen und Geschirrspülern sehr lange aufgeschoben werden kann. Aufgrund der damit verbundenen größeren Streuung werden zusätzlich die ursprünglichen Lastspitzen gleichmäßiger über einen größeren Zeitraum verteilt.

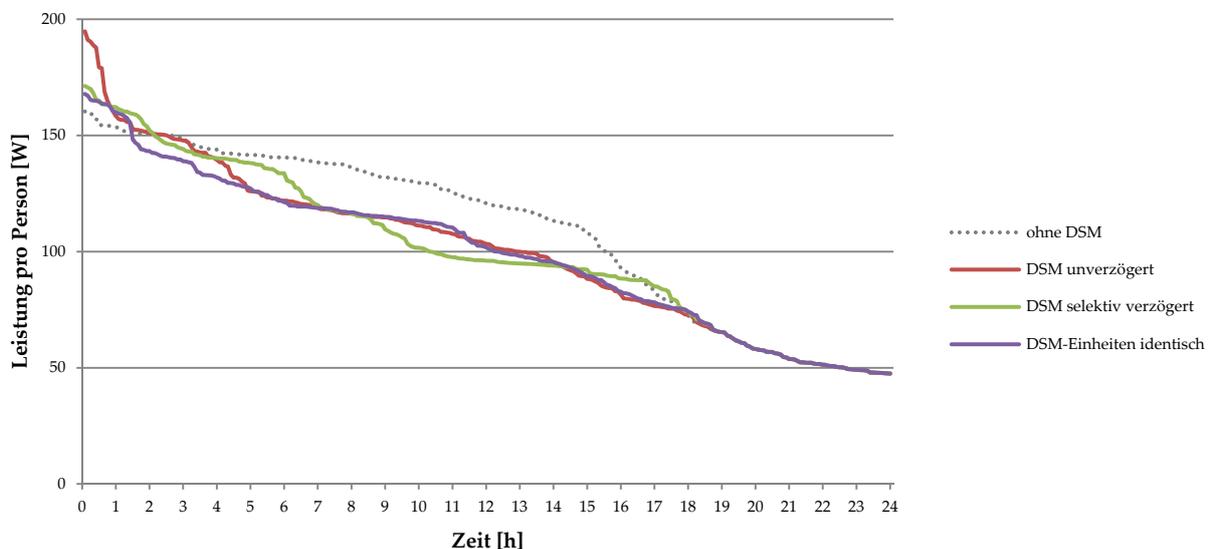


Diagramm 5.27: Lastdauerlinien der Gesamtleistung für das Szenario „langer Frequenzeinbruch“

Anhand der Lastdauerlinien in Diagramm 5.27 kann man die deutliche Reduzierung der aufgenommenen Leistung im Vergleich zu dem Gerätekollektiv ohne DSM erkennen.

Dass das DSM-System „selektiv verzögert“ für das hier betrachtete Szenario das günstigste Verhalten aufweist, kann sehr gut Diagramm 5.28 entnommen werden. Beinahe 12 Stunden kann dieses System eine Leistungsreduktion erzielen, zwei Stunden mehr als die beiden anderen Systeme.

Dieses Verhalten wird jedoch mit einem starken Eingriff in das Verhalten von Waschmaschinen und Geschirrspülern erreicht, deren Programm im Mittel um fünf Stunden verschoben wird. Ob dies für die Benutzer erträglich ist, muss bezweifelt werden.

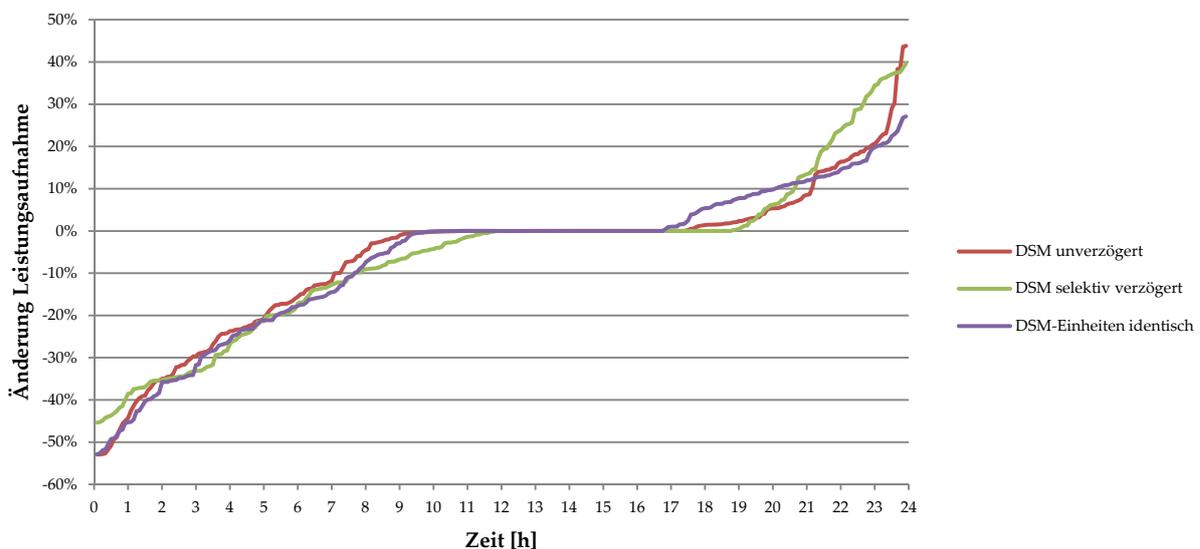


Diagramm 5.28: Dauerlinien der relativen Änderung der Gesamtleistung für das Szenario „langer Frequenzeinbruch“

5.4. Reaktion auf Frequenzverlauf bei einer Großstörung

Bis jetzt wurden die Frequenzverläufe für die Simulation als einfache, lineare Verläufe angenommen. Im folgenden Szenario soll die Netzfrequenz jedoch den Verlauf bei einer Großstörung im europäischen Übertragungsnetz nachbilden.

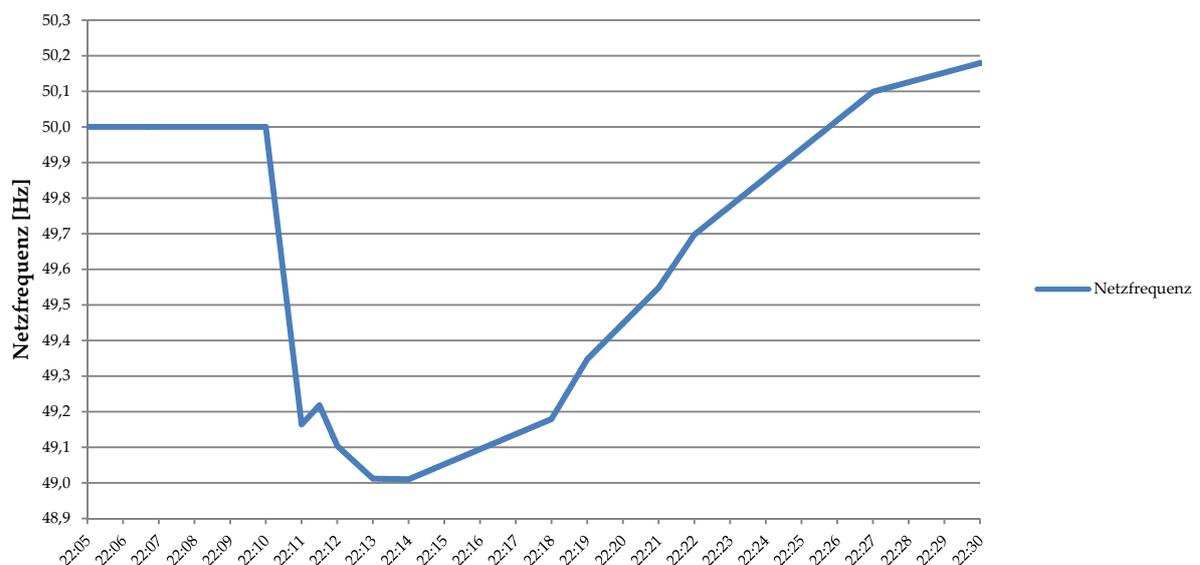


Diagramm 5.29: Näherung des Verlaufes der Netzfrequenz am 4. November 2006 in der Zeit von 22:05 bis 22:30 Uhr MEZ; Ursprungsdaten aus Messungen im Ruhrgebiet (Area 1) [19]

Dazu wurde die Großstörung, welche sich am 4. November 2006 ereignete, herangezogen. Genauere Informationen zu dieser können [4] entnommen werden. Diagramm 5.29 zeigt einen Ausschnitt aus den für dieses Szenario verwendeten Frequenzdaten, die dem realen Frequenzverlauf angenähert wurden. Um 22:30 Uhr wird der höchste Wert der Frequenz mit 50,19 Hz erreicht. Im Szenario wird dann angenommen, dass die Frequenz sich in 45 Minuten wieder auf den Wert von 50 Hz normalisiert (siehe Frequenzverlauf in Diagramm 5.30).

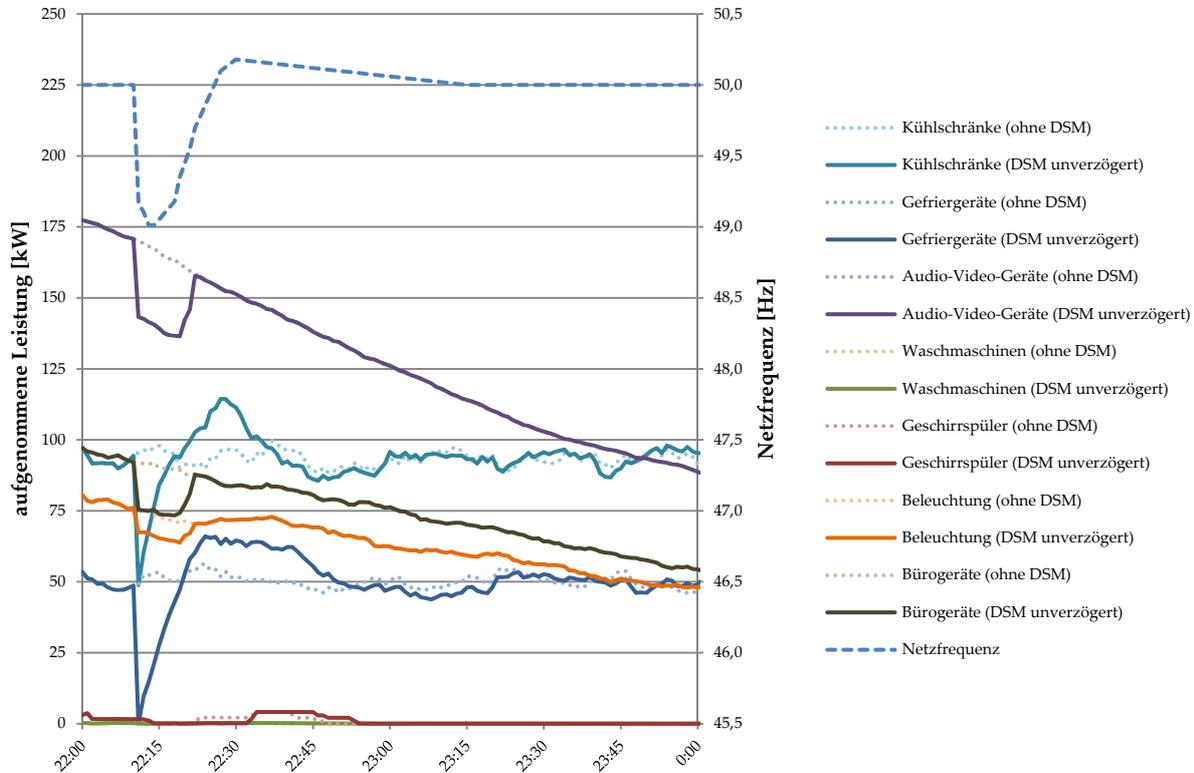


Diagramm 5.30: Ausschnitt aus dem Simulationsergebnis für das angegebene Szenario und dem DSM-System „unverzögert“

Für eine bessere Darstellung wurde die Simulation in Minutenschritten durchgeführt. Das Verhalten der einzelnen Gerätegruppen des Systems „DSM unverzögert“ ist in Diagramm 5.30 dargestellt.

Da die untersuchte Frequenzstörung sehr spät am Abend stattfand, sind an der Gesamtleistung fast keine Waschmaschinen und Geschirrspüler beteiligt. Den größten Teil der Leistungsreduktion können die Kühlgeräte zur Verfügung stellen. Die Gerätegruppen Audio-Video-Geräte, Beleuchtung und Bürogeräte beteiligen sich gemäß ihrem Potential.

Man kann erkennen, dass es beim DSM-System „unverzögert“ zu einer raschen Reaktion auf den Frequenzeinbruch und einer ebenso raschen Rückkehr zur normalen Betriebsweise bei der Erholung der Netzfrequenz kommt. Da die Dauer der Leistungsreduktion relativ kurz ist, kommt es bei den Kühlgeräten zu keiner stark erhöhten Leistungsaufnahme. Nach ca. einer Stunde entspricht die Leistungsaufnahme wieder jener ohne DSM (siehe Diagramm 5.32).

Im Gegensatz dazu erfolgt beim DSM-System „Einheiten identisch“ zwar ebenso eine schnelle Reaktion auf den Frequenzeinbruch, die Rückkehr zum normalen Betrieb wird aber deutlich hinausgezögert (siehe Diagramm 5.31 und Diagramm 5.32).

Das DSM-System „selektiv verzögert“ liegt mit seiner Leistungsaufnahme zwischen den beiden Anderen. Das ist durch den Parameterwert von `Time_Delay_Restore_Op` der Kühlgeräte zu begründen, der genau die Hälfte des Wertes des Systems „Einheiten identisch“ beträgt (siehe Tabelle 5.1).

Durch die längere Dauer des Zeitraumes, in dem die Solltemperatur erhöht bleibt, kommt es auch zu einer vermehrten Leistungsaufnahme der Kühlgeräte, wenn diese wieder ihre normale Solltemperatur einstellen. Da bei den DSM-Systemen „selektiv verzögert“ und „Einheiten identisch“ diese Zeitpunkte

für die einzelnen Geräte über einen größeren Zeitraum verteilt werden, wird verhindert, dass die Erhöhung zu hoch wird. Dennoch kann man diese anhand der relativen Änderung der Gesamtleistung in Diagramm 5.33 deutlich ausmachen. Sie fällt sogar höher aus als bei „DSM unverzögert“. In Diagramm 5.33 kann man ebenso gut erkennen, dass beim DSM-System „selektiv verzögert“ die Verzögerung nach zwei Stunden, bei „Einheiten identisch“ nach vier Stunden beendet wird, da hier die Erhöhungen der Leistungsaufnahme abklingen.

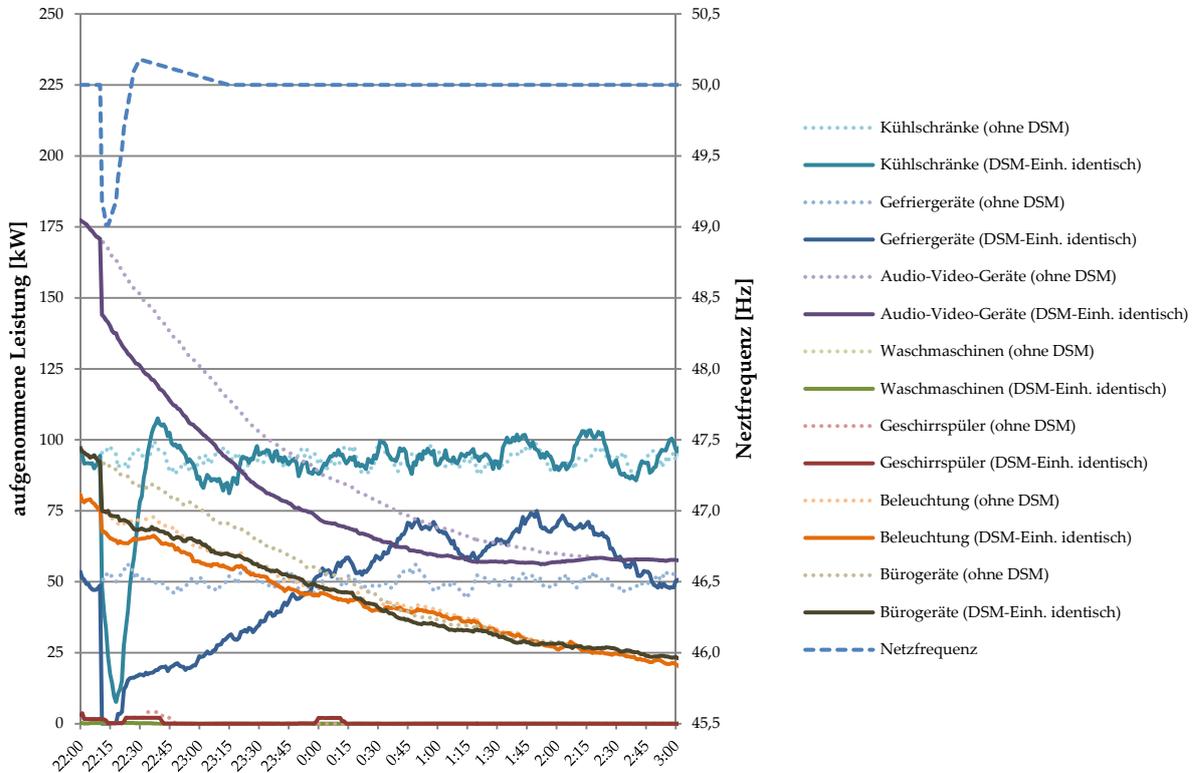


Diagramm 5.31: Ausschnitt aus dem Simulationsergebnis für das angegebene Szenario und dem DSM-System „Einheiten identisch“

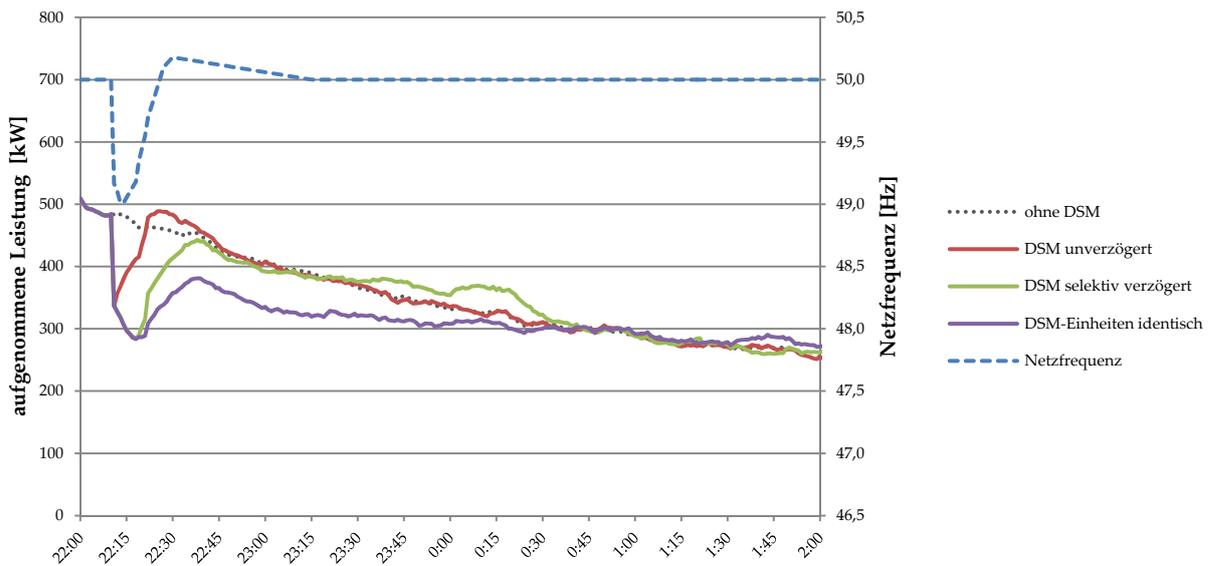


Diagramm 5.32: Vergleich der Gesamtleistungsaufnahme für die verschiedenen DSM-Systeme für das Szenario „Großstörung Europa 4.11.2006“

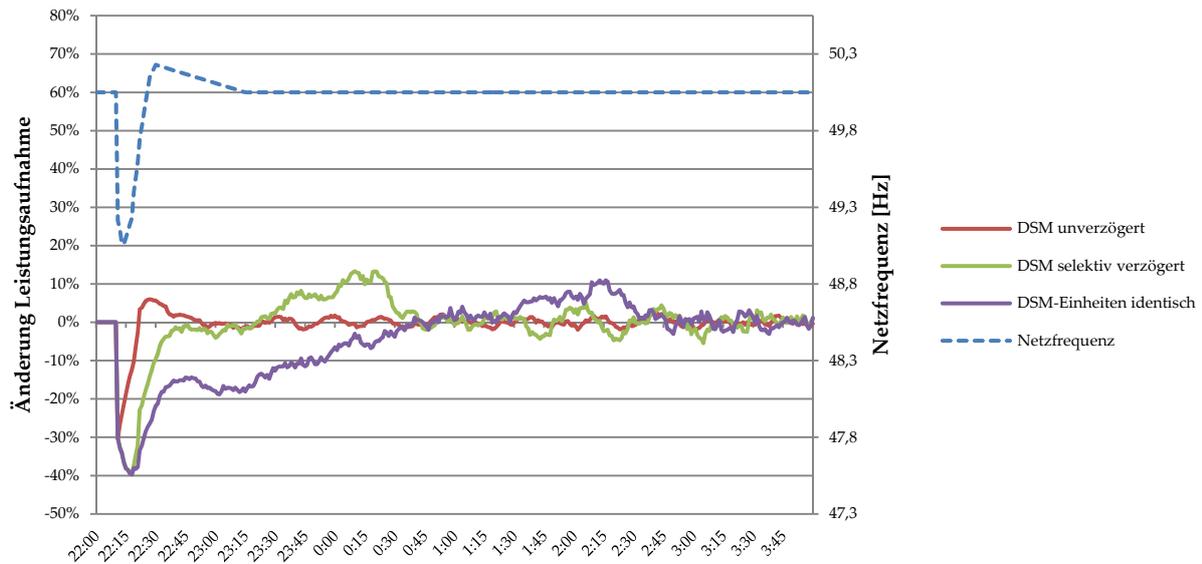


Diagramm 5.33: relative Änderung der Gesamtleistung bezogen auf die Leistungsaufnahme ohne DSM für das Szenario „Großstörung Europa 4.11.2006“

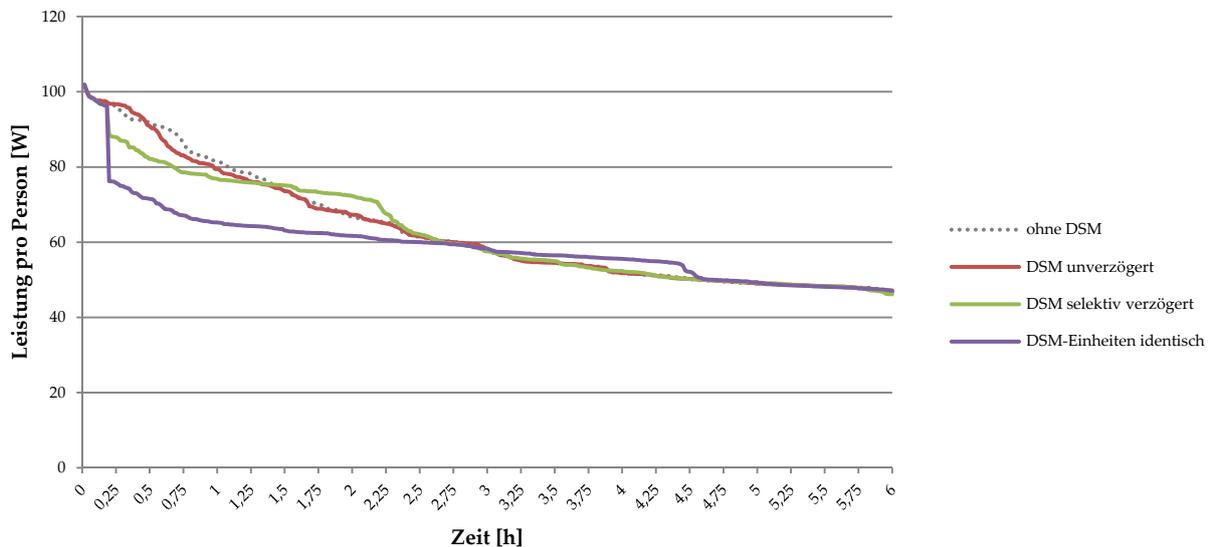


Diagramm 5.34: Lastdauerlinien der Gesamtleistung für das Szenario „Großstörung Europa 4.11.2006“

In Diagramm 5.34 und Diagramm 5.35 sind noch die Dauerlinien für die Gesamtleistung und die relative Änderung der Leistungsaufnahme dargestellt. Sie zeigen ein ähnliches Bild wie beim Szenario „kurzer Frequenzeinbruch“, da die untersuchte Großstörung einen ähnlichen Frequenzverlauf wie dieses aufweist.

Das DSM-System „Einheiten identisch“ sorgt für eine längere Leistungsreduktion als eigentlich notwendig wäre. Eine erhöhte Leistungsaufnahme wird vermieden. Gerade bei einer Netzstörung könnte dieses Verhalten aber von großem Vorteil sein, da so plötzliche Lastwechsel bei der Erholung des Netzes, welche dieses wieder stark belasten würden, nicht auftreten.

Die erzielte, kurzfristige Leistungsreduktion ist bei allen Systemen mit bis zu 40% bereits beachtlich. Um einen Black-Out zu vermeiden könnten aber alle Verbraucher noch einen höheren Beitrag leisten. So könnte z.B. die DSM-Einheit bei hohen und schnellen Frequenzeinbrüchen eine Art „Notprogramm“ starten, in dem deutlich mehr Verbraucher vom Netz getrennt werden. Auch könnten in diesem Modus andere Verzögerungszeiten gültig sein, um der Erholung des Netzes mehr Zeit zu lassen. Zu diesem Zweck wurde die `DSM_Device`-Instanz mit zwei Ausgaben ausgestattet (siehe Abschnitt 2.3.2) um so eine differenzierte Reaktion zu ermöglichen. Eine Implementierung dieser wurde jedoch noch nicht durchgeführt.

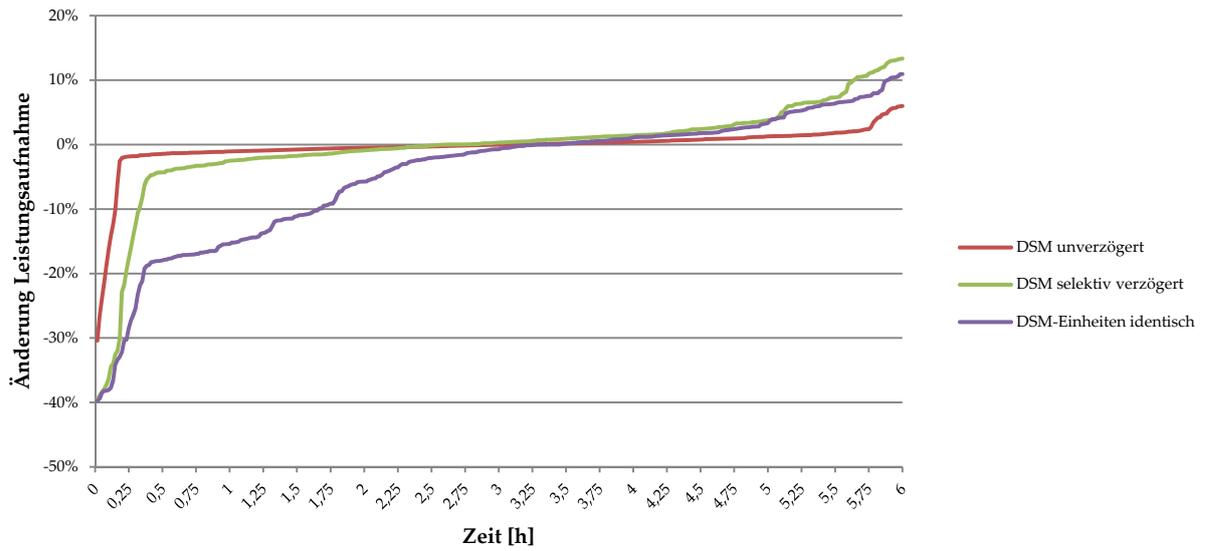


Diagramm 5.35: Dauerlinien der relativen Änderung der Gesamtleistung für das Szenario „Großstörung Europa 4.11.2006“

6. Schlussfolgerungen

Ziel dieser Arbeit ist es zu zeigen, wie ein möglichst simples DSM-System aussehen könnte, das ohne zusätzliche Kommunikation zwischen den Verbrauchern auskommen soll. Im Kapitel 1 wurden einige Fragen angeführt, die es in diesem Zusammenhang zu klären gilt. Mit Hilfe des erstellten Simulationsprogramms und dem damit entwickelten Simulationsmodell wurde versucht, erste Antworten auf diese Fragen zu finden.

Die Einbindung des DSM-Chips in die Geräte soll auf einfache Weise mit Hilfe einer definierten und eventuell sogar genormten Schnittstelle erfolgen. So muss die DSM-Einheit keine gerätespezifischen Aufgaben erfüllen und kann für jedes beliebige Gerät gleich ausgeführt werden. Dem Gerät wird über die Schnittstelle der aktuelle Netzzustand mitgeteilt, woraufhin dieses im Rahmen seiner Möglichkeiten reagiert. Ist die DSM-Einheit günstig in der Herstellung, kann so jedes beliebige Gerät für DSM herangezogen werden.

Dazu müssen jedoch die Geräte die Signale der DSM-Einheit verstehen und umsetzen können. So müssen z.B. bei Kühlgeräten und allen mit Thermostat gesteuerten thermischen Speichern entsprechende Möglichkeiten geschaffen werden, dass diese Geräte ihre Solltemperatur verändern können. Beleuchtungskörper müssen dimmbar sein, Audio-Video-Geräte eine Leistungsreduzierung durchführen können. Geht man von einem immer höher werdenden Einsatz von Microcomputern zur Steuerung von Elektrogeräten aus, können die hier formulierten Anforderungen sicherlich von einer Mehrzahl der Geräte erfüllt werden. Nur muss eine entsprechende Realisierung durch den Hersteller erfolgen.

Mit Hilfe erster Simulationen konnten bereits Eigenschaften von unterschiedlichen DSM-Systemen untersucht werden. Die in dieser Arbeit untersuchten DSM-Systeme sind nur ein erster Ansatz, um auch die Leistungsfähigkeit des entwickelten Simulationsprogramms zu demonstrieren. Dennoch konnten einige Erkenntnisse aus den Ergebnissen gewonnen werden.

So ist die Genauigkeit der Frequenzmessung nicht von großer Bedeutung. Auch hat die Streuung von gewissen Parametern kaum Einfluss auf die Änderung der Leistungsaufnahme, z.B. bei der maximalen Dauer der Verschiebung des Programmstartes bei Waschmaschinen und Geschirrspülern.

Ein besonders einfaches Mittel, um Lastspitzen bei der Erholung der Netzfrequenz zu vermeiden, ist die Verzögerung der Rückkehr zum normalen Betrieb. Es zeigte sich weiters, dass ein damit ausgestattetes DSM-System, das auch noch identische Einheiten in allen Gerätetypen verwendet, bereits ein gutes Verhalten in Bezug auf die Leistungsaufnahme aufweist.

Es bleiben dennoch viele Fragen offen. Eine weitergehende Untersuchung dieser sprengt aber den Umfang dieser Arbeit. In den abschließenden Abschnitten soll kurz erläutert werden, welche Erweiterungen und Ergänzungen sich bereits als notwendig herausstellten, um eine erschöpfende Behandlung von DSM im Haushalt zu ermöglichen.

6.1. Erweiterungen der Software und des Simulationsmodells

Bei der Erstellung des Simulationsmodells wurde versucht, so weit wie möglich, ein großes Angebot an möglichen Funktionalitäten zu bieten, damit möglichst viele verschiedene DSM-Systeme realisiert werden können.

Bei der Implementierung dieser Funktionen und der Untersuchung von ersten DSM-Systemen und Szenarien wurde rasch die Notwendigkeit für weitere mögliche und interessante Funktionen eines DSM-Systems erkannt. Für weitergehende Untersuchungen könnten diese einfach in das Simulationsprogramm eingebaut werden. Im Folgenden sollen kurz einige dieser weiterführenden Funktionen vorgestellt werden, wobei bereits einige davon in anderen Abschnitten erwähnt wurden.

Das Simulationsmodell ist in seinem bisherigen Aufbau noch sehr grob. Durch eine feinere Unterteilung der untersuchten Gerätesektoren können bessere Ergebnisse auch bei kleinen Gerätekollektiven erreicht werden, beispielsweise bei der Beleuchtung oder bei den Bürogeräten. Um das Modell auch verlässlicher in Bezug der Gerätezusammensetzung zu machen, müssten auch neuere und genauere Daten über diese ermittelt und in das Modell eingebaut werden.

Das Simulationsprogramm ist derzeit hauptsächlich auf die Simulation eines Tages ausgelegt. Es besteht aber bereits die Möglichkeit, den Simulationszeitraum beliebig zu verlängern. Dazu müsste aber auch das Simulationsmodell erweitert werden, um z.B. den unterschiedlichen Verbrauch an verschiedenen Wochentagen und zu verschiedenen Jahreszeiten zu berücksichtigen.

Die Parameterfindung für die Geräteparameter des Simulationsmodells ist derzeit noch etwas umständlich realisiert. Am günstigsten wäre hier eine automatische Parameterfindung mit Hilfe bereits vorhandener Daten wie z.B. Lastgänge.

Neben der Erweiterung des Simulationsmodells um die anderen, bisher nicht untersuchten Gerätegruppen im Haushalt (Kochen, Warmwasser, Umwälzpumpe, Trocknen) könnten auch andere Gerätegruppen aufgenommen werden, die zukünftige, neuartige Verbraucher darstellen sollen, wie z.B. Elektroautos. Auch könnten dezentrale Erzeugungsanlagen eines Haushalts (wie Photovoltaik-Anlagen, Biomasseheizungen mit Stromauskopplung, usw.) in das Simulationsprogramm eingebaut werden um deren Potential für eine „Leistungsreduktion“ zu untersuchen. Dazu müssten neue Geräteklassen definiert werden.

Neben den Geräteklassen könnten neue Verbrauchergruppen definiert werden, um z.B. einzelne Haushalte nachzubilden. So könnte dann die Leistungsaufnahme von verschiedenen Haushalten genauer betrachtet werden.

Bei den bisher gemachten Untersuchungen wurde nur die Reaktion der Verbraucher auf eine Frequenzreduktion betrachtet. Es kann aber auch eine Frequenzerhöhung Informationen zur Erzeugungssituation liefern. So könnte z.B. damit ein „aufladen“ von Energiespeichern initiiert werden (z.B. Verringerung der Soltemperatur in Kühlgeräten oder Aufheizen von Warmwasserboilern) um eine erhöhte Erzeugung abzufangen. Dazu müsste die Schnittstelle der DSM-Einheit erweitert werden, z.B. um ein Signal „Charge“.

Bei Frequenzeinbrüchen könnte auch unterschieden werden, ob diese z.B. langsam oder sehr schnell erfolgen und daraus folgend eine schwächere oder stärkere Reaktion des Systems zu erreichen. Die hier vorgestellte Schnittstelle ist in den Grundzügen bereits darauf vorbereitet.

Neben diesen Erweiterungen muss natürlich auch das Simulationsprogramm selbst den Anforderungen entsprechend erweitert werden, auch um eine bessere Handhabung zu ermöglichen.

6.2. Weiterführende Untersuchungen

Neben den ersten, bereits durchgeführten Untersuchungen an DSM-Systemen sind noch viele weitere notwendig, um Potentiale und Auswirkungen richtig einschätzen zu können.

So wurden bisher die Frequenzdaten für die hier untersuchten Simulationen in einem Zeitabstand von 30 Sekunden dem Programm und damit den DSM-Einheiten übermittelt. Inwieweit diese Genauigkeit ausreicht, muss noch geklärt werden. Auch könnte der Einfluss der Dauer der Zeiträume zwischen den Messungen der Frequenz untersucht werden, um festzustellen, in welchen Zeitabständen eine DSM-Einheit diese zumindest durchführen muss.

Alle DSM-Einheiten führen die Frequenzmessung derzeit „gleichzeitig“ aus. Mit genaueren Frequenzdaten könnte untersucht werden, inwieweit sich dieser Umstand auf die Leistungsreduktion auswirkt. In der Realität wäre so eine Bedingung an das DSM-System nur sehr schwer zu erfüllen, weshalb gezeigt werden soll, dass eine zeitliche Verteilung der Messungen zu keiner Funktionsbeeinträchtigung führt.

Im Abschnitt 2.3.1.2.1 wurde bereits kurz auf verrauschte Frequenzdaten eingegangen. Wie sich die Behandlung dieser auswirkt und welche Filterung notwendig ist, um ein funktionierendes DSM-System zu erhalten, kann mit den bereits implementierten Funktionen untersucht werden.

Bei einer Leistungsreduktion werden jene Verbraucher herangezogen, die zufällig zu diesem Zeitpunkt in Betrieb sind. Auf längere Sicht gesehen müsste jedoch ein Ausgleich geschaffen werden, so dass nicht immer die gleichen Geräte ihre Leistung reduzieren. So kann es z.B. bei einem Gefrierschrank zu ungünstigen Auswirkungen auf die Haltbarkeit der Lebensmittel kommen, wenn dieser ständig mit erhöhter Soll-Temperatur betrieben wird, andere Gefriergeräte sich trotz DSM aber nicht an der Leistungsredukti-

on beteiligen. So soll untersucht werden, inwieweit verschiedene DSM-Systeme einzelne Geräte ungünstig belasten und wie dies verhindert werden kann.

Auch das Einführen verschiedener Prioritätsgruppen wurde noch keiner genaueren Betrachtung unterzogen.

Die DSM-Systeme wurden bisher nur an einem einzigen Gerätekollektiv getestet. Szenarien mit unterschiedlichen Personenzahlen könnten z.B. Aufschluss über das Verhalten von einzelnen Geräten oder Haushalten geben.

Die in den Abschnitten 6.1 und 6.2 angeführten offenen Teilaspekte zeigen, dass das Thema „Umsetzung von Demand Side Management“ ein sehr komplexes ist und eine weitere erschöpfende Behandlung einen hohen Aufwand bedeutet. In dieser Arbeit wurde versucht, die Basis für diese zu schaffen. Darauf aufbauend sollen bald weitere mögliche Umsetzungsmöglichkeiten getestet und bewertet werden können.

Anhang

A Literaturverzeichnis

- [1] Christoph Groß, **Power Demand Side Management: Potentiale und technische Realisierbarkeit im Haushalt**, Diplomarbeit, Institut für Elektrische Anlagen und Energiewirtschaft, TU Wien, 2008.
- [2] Günther Brauner, Georg Pöppl, Dietmar Tiefgraber, **Verbraucher als virtuelles Kraftwerk: Potentiale für Demand Side Management in Österreich im Hinblick auf die Integration von Windenergie**, Institut für Elektrische Anlagen und Energiewirtschaft, TU Wien, 2006.
- [3] Alfred Einfalt, **Stochastisches Energiemanagement von Kleinverbrauchern**, Diplomarbeit, Institut für Elektrische Anlagen und Energiewirtschaft, TU Wien, 2006.
- [4] **Final Report: System Disturbance on 4th November 2006**, UCTE, Jan. 2007.
- [5] Michael Gammer, **Simulation der Großstörung in Italien**, Diplomarbeit, Institut für Elektrische Anlagen und Energiewirtschaft, TU Wien, 2005.
- [6] Christian Bendel, David Nestle, Jan Ringelstein, **Wirtschaftliche und technische Integration verteilter Erzeugung in der Praxis - Feldtest im Projekt DINAR**, Institut für Solare Energieversorgungstechnik ISET (Kassel), 22. Symposium Photovoltaische Solarenergie, Bad Staffelstein, Deutschland, 2007.
- [7] Christoph Groß, Günther Brauner, **Power Demand Side Management – Potentialabschätzung im Haushalt**, TU Wien, Institut für Elektrische Anlagen und Energiewirtschaft, 6. Internationale Energiewirtschaftstagung an der TU Wien, Jan. 2009.
- [8] Friederich Kupzog, **Self-controlled Exploitation of Energy Cost saving Potentials by Implementing distributed Demand Side Management**, TU Wien, Institut für Computertechnik, 4. International Conference on Industrial Informatics, Singapore, 2006, 2006.
- [9] Friederich Kupzog, **Frequency-responsive load management in electric power grids**, Dissertation, Institute of Computer Technology, TU Wien, 2008.
- [10] D. Hammerstrom, **Pacific Northwest Gridwise™ Testbed Demonstration Projects: Part II. Grid Friendly™ Appliance Project**, Project Report 2007, U.S. Department of Energy, Oktober 2007.
- [11] digitalSTROM.org, **digitalSTROM beziehen**, Adresse: <http://www.digitalstrom.org/beziehen/beziehen.html>, Zuletzt geprüft am 23.11.2010.
- [12] digitalSTROM.org, **Poster digitalSTROM-System I**, Adresse: http://www.digitalstrom.org/fileadmin/USERS/ABLAGE/download/pdfs/20100105_digitalSTROM_System2.pdf, Zuletzt geprüft am 23.11.2010.
- [13] Gunnar Bärwaldt, Michael Kurrat, **Auswirkungen von dynamischen Haushaltsstromtarifen auf der Basis stochastischer Haushaltslastprofile**, Technische Universität Braunschweig, Institut für Hochspannungstechnik und Elektrische Energieanlagen, Februar 2008.
- [14] Barvir Sanosian, **Simulationssystem zur Untersuchung elektrischer Netze von kleinen Siedlungen**, Institut für Elektrische Anlagen und Energiewirtschaft, TU Wien, 2009.
- [15] Europäisches Parlament und Rat, **Richtlinie 2005/32/EG des Europäischen Parlaments und Rates vom 6.Juli 2005 zur Schaffung eines Rahmens für die Festlegung von Anforderungen an die umweltgerechte Gestaltung energiebetriebener Produkte und zur Änderung der Richtlinie 92/42/EWG sowie der Richtlinien 96/57/EG und 2000/55/EG des Europäischen Parlaments und Rates, ABl. L 191 vom 22.7.2005: Ökodesign-Richtlinie**. 6.Juli 2005, 2005.
- [16] guenstiger-drucken.de, **Der Druckermarkt wächst: News von guenstiger-drucken.de am 02.September 2009**. Adresse: <http://guenstiger-drucken.de/der-druckermarkt-wachst-886/>, Zuletzt geprüft am 23.11.2010.

- [17] Matthias Parbel, **Drucker-Markt im Aufwind: Preise und Verkaufszahlen steigen wieder**, Adresse: <http://www.heise.de/newsticker/meldung/Drucker-Markt-im-Aufwind-Preise-und-Verkaufszahlen-steigen-wieder-993203.html>, Zuletzt geprüft am 23.11.2010.
- [18] Thomas Siebenhüner, **Demand-Side-Management von Kleinverbrauchern**, Diplomarbeit, Institut für Elektrische Anlagen und Energiewirtschaft, TU Wien, 2006.
- [19] Wikipedia, **Netzfrequenz 11.04.2006 Area 1**: *Messung selbst durchgeführt, Bild selbst erstellt*, Adresse: http://de.wikipedia.org/w/index.php?title=Datei:Netzfrequenz_20061104_Area_1.jpg&filetimestamp=20061112123742, Zuletzt geprüft am 23.11.2010.

B Abbildungsverzeichnis

B1 Abbildungen

Abbildung 1.1: Design der DSM-Interface-Einheit IRON Box (aus [9: S.104])	2
Abbildung 1.2: Der „GridFriendly™ Appliance Controller“ (aus [9: S.25]).....	3
Abbildung 1.3: Kommunikationsstruktur des BEMI, ©ISET (aus [6: S.14]).....	3
Abbildung 1.4: Aufbau des digitalSTROM® Systems (aus [12])	4
Abbildung 2.1: grobe Übersicht über die Struktur der implementierten Geräteklassen.	12
Abbildung 2.2: Modell für die Klasse <code>Thermal_Storage</code> (nach [9: S.47])	13
Abbildung 2.3: Modell eines thermischen Speichers, dargestellt als elektrische Schaltung (nach [9: S.46])14	
Abbildung 2.4: Schema der Erzeugung der Geräteinstanzen und der <code>DSM_Device</code> -Instanzen.....	51
Abbildung 2.5: Aufbau der <code>DSM_Device</code> -Instanz sowie Ablauf der Ermittlung der aufgenommenen Leistung für einen Simulationsschritt (ohne <code>look_back</code> -Funktion).....	52
Abbildung 2.6: erzeugte Dateien nach einem Simulationsdurchlauf	53
Abbildung 2.7: Beispiel für eine Ausgabe der Simulationsergebnisse durch das Simulationsprogramm ..	54
Abbildung 2.8: Schema des Aufbaus der Simulationssoftware.....	55
Abbildung 2.9: Hauptfenster des Simulationsprogrammes.....	55
Abbildung 2.10: Fenster mit den Einstellungen für die Durchführung von Simulationsreihen.....	57

B2 Diagramme

Diagramm 2.1: Auswirkung bei der Behandlung negativer Parameterwerte am Beispiel des Parameters <code>Time_Delay_Restore_Op = 100 min, Std.Dev. = 50%</code>	11
Diagramm 2.2: Vergleich des gleichen thermischen Speichers (Kühlgerät) mit unterschiedlichen Werten für Δt	15
Diagramm 2.3: Vergleich des gleichen thermischen Speichers (Kühlgerät) mit unterschiedlichen Werten für Δt und Inaktivität des Geräts von 0:30 bis 12:00 Uhr	16
Diagramm 2.4: Vergleich der aufgenommenen Leistung und der Temperatur eines thermischen Speichers (Kühlgerät) für verschiedene globale Simulationsauflösungen.	16
Diagramm 2.5: Auswirkung der Änderung verschiedener Parameter (Verdopplung) der Klasse <code>Thermal_Storage</code> am Beispiel eines Kühlgerätes.....	17
Diagramm 2.6: Auswirkung der Änderung verschiedener Parameter (Halbierung) der Klasse <code>Thermal_Storage</code> am Beispiel eines Kühlgerätes.....	18
Diagramm 2.7: Erreichen der Übereinstimmung der Ergebnisse des fein und grob zeitlich aufgelösten thermischen Modells mit Hilfe von Variierung von R_{th}	19
Diagramm 2.8: Illustrierung zum Beispiel für die Umwandlung eines Einsatzplanes	21
Diagramm 2.9: Vergleich des Simulationsergebnisses der gleichen Geräteinstanzen bei verschiedenen Auflösungen	22
Diagramm 2.10: Auswirkung eines Stand-by-Verbrauchs von 20 W bei dem gleichen Einsatzplan von Diagramm 2.8.	23
Diagramm 2.11: Beispiel einer Instanz von <code>Periodic_Operation</code> mit den Parameterwerten aus Tabelle 2.8	24
Diagramm 2.12: Zusammenspiel von Einsatzplänen für verschiedene Benutzerzahlen bei der Klasse <code>Probable_Operation</code> . Als Parameter wurden für das obige Beispiel die Werte von Tabelle 3.8 auf Seite 61 herangezogen (Audio-Video-Geräte).....	25
Diagramm 2.13: Darstellung der Auswirkung der Parameter in der Klasse " <code>Probable_Operation</code> "	26
Diagramm 2.14: Darstellung der Lastkurven aus dem Beispiel angegeben in Tabelle 2.13	29
Diagramm 2.15: Verschiebung der Lastkurven bei kurz aufeinander folgenden Startzeitpunkten mit Lastkurven angegeben in Tabelle 2.13	30
Diagramm 2.16: Auswirkung verschiedener Standardabweichungen des Frequenzschwellwertes auf die DSM-Funktion. Daten von 1.000 Verbrauchern	33
Diagramm 2.17: Auswirkungen verschiedener Breiten der Reaktionshysterese bei verrauschten Frequenzdaten, Daten von 1.000 Verbrauchern	34
Diagramm 2.18: Detail aus Diagramm 2.17	34

Diagramm 2.19: Auswirkungen verschiedener Zeiträume der laufenden Mittelwertbildung bei verrauschten Frequenzdaten, Daten von 1.000 Verbrauchern.....	35
Diagramm 2.20: Auswirkungen des Parameters <code>Time_Delay_Restore_Op</code> auf die Leistungsreduktion.	36
Diagramm 2.21: Auswirkung verschiedener Standardabweichungen der Frequenzschwellwerte auf die DSM-Funktion beim Modus <code>Frequency_Response_Priority</code> . Parameter nach Tabelle 2.17, Daten von 1.000 Verbrauchern	38
Diagramm 2.22: <code>Turn_Off</code> bei einem Gerät der Klasse <code>Probable_Operation</code> mit Stand-by-Verbrauch.	40
Diagramm 2.23: <code>Turn_Off</code> bei einem Gerät der Klasse <code>Thermal_Storage</code> (Kühlgerät). In der Zeit von 0:30 bis 12:00 bricht die Frequenz um 2 Hz ein (ähnlich Diagramm 2.22, nicht eingezeichnet).....	41
Diagramm 2.24: <code>Turn_Off</code> bei einem Gerät der Klasse <code>Loadcurve_Operation</code>	41
Diagramm 2.25: <code>Turn_Off_Stand_by</code> bei einem Gerät der Klasse <code>Probable_Operation</code>	42
Diagramm 2.26: <code>Reduce_Input_Power</code> bei einer Instanz der Klasse <code>Probable_Operation</code>	43
Diagramm 2.27: <code>Change_Temp_Set</code> bei einer Instanz von <code>Thermal_Storage</code> (Kühlgerät).....	44
Diagramm 2.28: Auswirkung unterschiedlicher Simulationsauflösungen auf das Ergebnis bei <code>Check_former_Frequ_Data = 0</code>	44
Diagramm 2.29: Auswirkung unterschiedlicher Simulationsauflösungen auf das Ergebnis bei <code>Check_former_Frequ_Data = 1</code>	45
Diagramm 2.30: Darstellung der Auswirkungen verschiedener Einstellungen von <code>Postpone_Start</code> der gleichen Geräteinstanz.....	46
Diagramm 2.31: Ausgangssituation für Diagramm 2.32; ca. 150 Instanzen des Gerätetyps <code>Loadcurve_Operation</code> mit den Parametern aus Tabelle 3.19 und einer Simulationsauflösung von 5 Minuten	47
Diagramm 2.32: Auswirkung unterschiedlicher Simulationsauflösungen auf das Simulationsergebnis... ..	48
Diagramm 2.33: Einfluss der Option <code>Consider_non_stop_parts</code>	49
Diagramm 2.34: Einfluss des Parameters <code>Time_Postpone_max</code> auf die Output-Funktion <code>Pause_Programm</code>	49
Diagramm 3.1: Summenlastganglinie über die einzelnen Sektoren (nach [1: S.96])	59
Diagramm 3.2: Ergebnis von vier Simulationsdurchläufen mit ca. 10.000 aktiven Bürogeräten mit Abweichungen von den Vergleichsdaten	62
Diagramm 3.3: Ergebnis von vier Simulationsdurchläufen mit ca. 10.000 aktiven Audio-Video-Geräten mit Abweichungen von den Vergleichsdaten	65
Diagramm 3.4: Ergebnis von vier Simulationsdurchläufen mit ca. 10.000 aktiven Beleuchtungskörpern mit Abweichungen von den Vergleichsdaten	66
Diagramm 3.5: Wirkleistungsaufnahme eines Kühlschranks „Whirlpool ART 455/4“. Ausschnitt aus der Messung (5h).....	69
Diagramm 3.6: Vergleich der Auswirkungen der verschiedenen Parameterwerte und Berechnungsmethoden im thermischen Modell des Durchschnittskühlschranks; untere Kurven: Aufgenommene Leistung [kW]; obere Kurven: Temperatur im Kühlschrank [°C].....	69
Diagramm 3.7: Vergleich der thermischen Parameter des Durchschnittskühlschranks für die verschiedenen Simulationsauflösungen. Parameter gem. Tabelle 3.12	70
Diagramm 3.8: Ergebnis von vier Simulationsdurchläufen mit ca. 10.000 aktiven Kühlschränken mit Abweichungen von den Vergleichsdaten	72
Diagramm 3.9: Vergleich der Auswirkungen der verschiedenen Parameterwerte und Berechnungsmethoden im thermischen Modell des Durchschnittsgefriergerätes; untere Kurven: Aufgenommene Leistung [kW]; obere Kurven: Temperatur im Kühlschrank [°C].....	73
Diagramm 3.10: Vergleich der thermischen Parameter des Durchschnittskühlschranks für die verschiedenen Simulationsauflösungen. Parameter gem. Tabelle 3.15	74
Diagramm 3.11: Ergebnis von vier Simulationsdurchläufen mit ca. 10.000 aktiven Gefriergeräten mit den Abweichungen von den Vergleichsdaten	75
Diagramm 3.12: Lastgang einer Waschmaschine „AEG Lavamat 64618“ mit dem Programm „30min Super-schnell (30°)“ sowie daraus ermittelte Lastkurve (30% Abweichung von vorangegangenen Wert) und weitere mögliche Lastkurven (strichliert)	76

Diagramm 3.13: Lastgang einer Waschmaschine „AEG Lavamat 64618“ mit dem Programm „40°-60° Mix“ sowie daraus ermittelte Lastkurve (50% Abweichung von vorangegangenem Wert).....	76
Diagramm 3.14: Lastgang einer Waschmaschine „AEG Lavamat 64618“ mit dem Programm „60°“ sowie daraus ermittelte Lastkurve (30% Abweichung von vorangegangenem Wert).....	77
Diagramm 3.15: Ergebnis von vier Simulationsdurchläufen mit ca. 12.000 aktiven Waschmaschinen mit den Abweichungen von den Vergleichsdaten	79
Diagramm 3.16: Lastgang eines Geschirrspülers „Whirlpool ADG8527“ mit dem Programm „60°“ sowie daraus ermittelte Lastkurve (30% Abweichung von vorangegangenem Wert).....	80
Diagramm 3.17: Ergebnis von vier Simulationsdurchläufen mit ca. 10.000 aktiven Waschmaschinen mit den Abweichungen von den Vergleichsdaten	81
Diagramm 3.18: Ergebnis der Simulation der Leistungsaufnahme von Verbrauchern von 50.000 Personen	82
Diagramm 3.19: Abweichung des simulierten Gesamtverbrauchs von den Vergleichsdaten	83
Diagramm 5.1: Unbeeinflusste Leistungsaufnahme des für dieses Szenario verwendeten Verbraucherkollektivs.....	87
Diagramm 5.2: Simulationsergebnis für das angegebene Szenario mit einer Standardabweichung von Frequency_Level von 0%	88
Diagramm 5.3: Auswirkung verschiedener Standardabweichungen des Parameters Frequency_Level auf die Gesamtleistung	89
Diagramm 5.4: Änderung der Gesamtleistungsaufnahme bezogen auf die Leistungsaufnahme ohne Einsatz von DSM für unterschiedliche Standardabweichungen des Parameters Frequency_Level	89
Diagramm 5.5: Auswirkung unterschiedlicher Anteile von Geräten mit DSM auf die Gesamtleistung.....	90
Diagramm 5.6: Änderung der Gesamtleistungsaufnahme bezogen auf die Leistungsaufnahme ohne Einsatz von DSM für unterschiedlichen Anteil von Geräten mit DSM.....	90
Diagramm 5.7: Auswirkung unterschiedlicher DSM-Output-Moden und Einstellungen auf die Leistungsaufnahme der Waschmaschinen. Leseweise der Legende: DSM-Output-Mode; Time_Postpone_Start; Time_Postpone_max	91
Diagramm 5.8: Auswirkung unterschiedlicher Standardabweichungen der Parameter Time_Postpone_Start und Time_Postpone_max auf die Beeinflussung der Leistungsaufnahme von Waschmaschinen	91
Diagramm 5.9: Gleichverteilung der Parameter Time_Postpone_Start und Time_Postpone_max bei Waschmaschinen; Leseweise der Legende: DSM-Output-Mode; Time_Postpone_Start; Time_Postpone_max.....	92
Diagramm 5.10: Auswirkung unterschiedlicher Werte für den Parameter Time_Delay_Restore_Op bei Waschmaschinen; Leseweise der Legende: DSM-Output-Mode; Time_Delay_Restore_Op.....	93
Diagramm 5.11: Auswirkung unterschiedlicher DSM-Output-Moden und Einstellungen auf die Leistungsaufnahme der Geschirrspüler, Standardabweichung der Parameter 30%, Leseweise der Legende: DSM-Output-Mode; Time_Postpone_Start; Time_Postpone_max.....	94
Diagramm 5.12: Gleichverteilung der Parameter Time_Postpone_Start und Time_Postpone_max bei Geschirrspüler; Leseweise der Legende: DSM-Output-Mode; Time_Postpone_Start; Time_Postpone_max.....	94
Diagramm 5.13: Auswirkung unterschiedlicher Werte für den Parameter Time_Delay_Restore_Op bei Geschirrspülern.....	95
Diagramm 5.14: Einfluss des Parameters Time_Delay_Restore_Op mit unterschiedlichen Streuungen gemäß einer Standardverteilung auf die Leistungsaufnahme von Kühlschränken und Gefriergeräten.....	95
Diagramm 5.15: Einfluss des Parameters Time_Delay_Restore_Op mit Streuung gemäß einer Gleichverteilung auf die Leistungsaufnahme von Kühlschränken und Gefriergeräten.....	96
Diagramm 5.16: Simulationsergebnis für das betrachtete Szenario für Einsatz identischer DSM-Chips in allen Geräten, Time_Delay_Restore_Op = 240 min gleichverteilt; Time_Postpone_max = 120 min mit 30% Standardabweichung;	97
Diagramm 5.17: Vergleich der Gesamtleistungsaufnahme der exemplarischen DSM-Systeme für das Szenario „Verschiebung Lastspitzen“	99

Diagramm 5.18: relative Änderung der Gesamtleistung bezogen auf die Leistungsaufnahme ohne DSM für das Szenario „Verschiebung Lastspitzen“	99
Diagramm 5.19: Lastdauerlinien der Gesamtleistung für die verschiedenen DSM-Systeme und das Szenario „Verschiebung Lastspitzen“	100
Diagramm 5.20: Dauerlinien der relativen Änderung der Gesamtleistung für das Szenario „Verschiebung Lastspitzen“	100
Diagramm 5.21: Vergleich der Gesamtleistungsaufnahme für die verschiedenen DSM-Systeme für das Szenario „kurzer Frequenzeinbruch“	101
Diagramm 5.22: relative Änderung der Gesamtleistung bezogen auf die Leistungsaufnahme ohne DSM für das Szenario „kurzer Frequenzeinbruch“	102
Diagramm 5.23: Lastdauerlinien der Gesamtleistung für das Szenario „kurzer Frequenzeinbruch“	102
Diagramm 5.24: Dauerlinien der relativen Änderung der Gesamtleistung für das Szenario „kurzer Frequenzeinbruch“	103
Diagramm 5.25: Vergleich der Gesamtleistungsaufnahme für die verschiedenen DSM-Systeme für das Szenario „langer Frequenzeinbruch“	103
Diagramm 5.26: relative Änderung der Gesamtleistung bezogen auf die Leistungsaufnahme ohne DSM für das Szenario „langer Frequenzeinbruch“	104
Diagramm 5.27: Lastdauerlinien der Gesamtleistung für das Szenario „langer Frequenzeinbruch“	104
Diagramm 5.28: Dauerlinien der relativen Änderung der Gesamtleistung für das Szenario „langer Frequenzeinbruch“	105
Diagramm 5.29: Näherung des Verlaufes der Netzfrequenz am 4.November 2006 in der Zeit von 22:05 bis 22:30 Uhr MEZ; Ursprungsdaten aus Messungen im Ruhrgebiet (Area 1) [19]	105
Diagramm 5.30: Ausschnitt aus dem Simulationsergebnis für das angegebene Szenario und dem DSM-System „unverzögert“	106
Diagramm 5.31: Ausschnitt aus dem Simulationsergebnis für das angegebene Szenario und dem DSM-System „Einheiten identisch“	107
Diagramm 5.32: Vergleich der Gesamtleistungsaufnahme für die verschiedenen DSM-Systeme für das Szenario „Großstörung Europa 4.11.2006“	107
Diagramm 5.33: relative Änderung der Gesamtleistung bezogen auf die Leistungsaufnahme ohne DSM für das Szenario „langer Frequenzeinbruch“	108
Diagramm 5.34: Lastdauerlinien der Gesamtleistung für das Szenario „langer Frequenzeinbruch“	108
Diagramm 5.35: Dauerlinien der relativen Änderung der Gesamtleistung für das Szenario „langer Frequenzeinbruch“	109

C Tabellenverzeichnis

Tabelle 2.1: Beispiel für den Aufbau einer Parametertabelle „Parameters“. Die farbigen Markierungen dienen ausschließlich der besseren Übersicht.....	8
Tabelle 2.2: Zusammenfassung der Simulationseinstellungen.....	9
Tabelle 2.3: Übersicht der Properties der Klasse Device.....	12
Tabelle 2.4: Übersicht über die Methodenrümpfe, die in der Klasse Device definiert werden, und in allen Geräteklassen implementiert werden müssen.....	12
Tabelle 2.5: Übersicht der Properties der Klasse Thermal_Storage	17
Tabelle 2.6: Übersicht der Properties der Klasse Scheduled_Operation.....	20
Tabelle 2.7: Übersicht der Properties der Klasse Periodic_Operation.....	23
Tabelle 2.8: Parameterwerte für das Beispiel einer Instanz von Periodic_Operation (Auszug aus Parameterdatei).....	24
Tabelle 2.9: Parameter der Referenzkurve für Diagramm 2.13 (Auszug aus Parameterdatei)	26
Tabelle 2.10: Übersicht über die Variationen der verschiedenen Parameter für Diagramm 2.13	26
Tabelle 2.11: Übersicht der Properties der Klasse Loadcurve_Operation.....	28
Tabelle 2.12: Beispiel für eine Lastkurve.....	27
Tabelle 2.13: Beispiel für die Angabe mehrerer Lastkurven mit deren Auftrittshäufigkeit sowie der Definition von Programmteilen, die im laufenden Betrieb nicht unterbrochen werden sollen	28
Tabelle 2.14: Zusammenfassung der Parameter, Attribute, Moden und Ausgaben des Typs „DSM-Input“ der Klasse DSM_Device	32
Tabelle 2.15: Beispiel für die Definition der Verteilung der DSM-Input-Moden (Auszug aus Parameterdatei).....	32
Tabelle 2.16: Beispiel für Definition der Parameter für Frequency_Response_Simple (Auszug aus Parameterdatei).....	33
Tabelle 2.17: Definition von Prioritätsgruppen (Auszug aus Parameterdatei).....	37
Tabelle 2.18: Zusammenfassung der Parameter, Attribute, Moden und Ausgaben des Typs „DSM-Output“ der Klasse DSM_Device	39
Tabelle 2.19: Beispiel für eine Job-Listendatei in der neben unterschiedlichen Parameterdateien auch unterschiedliche Frequenzdaten definiert werden.	53
Tabelle 3.1: Bedarf an elektrischer Energie in Deutschland (2006) (aus [1: S.12])	60
Tabelle 3.2: Verkaufte Bürogeräte in Deutschland 2007 mit Aufschlüsselung auf Geräte pro Person. Aus [1: S.19]	61
Tabelle 3.3: Ermittlung der durchschnittlichen Anschlussleistung pro Person für Bürogeräte.....	62
Tabelle 3.4: Parameterwerte für Bürogeräte (Auszug aus Parameterdatei).....	62
Tabelle 3.5: Zusammenfassung von ermittelten Werten aus den Simulationsdurchläufen mit Bürogeräten von 10.300 Personen	63
Tabelle 3.6: Stand-by-Verbrauch im Sektor TV-Audio, aus [1: S.38].....	64
Tabelle 3.7: Ermittlung der Anschlussleistung pro Person für Audio-Video-Geräte	64
Tabelle 3.8: Parameterwerte für Audio-Video-Geräte (Auszug aus Parameterdatei).....	64
Tabelle 3.9: Zusammenfassung von ermittelten Werten aus den Simulationsdurchläufen mit Audio-Video-Geräten von 13.600 Personen	65
Tabelle 3.10: Parameterwerte für die Beleuchtung (Auszug aus Parameterdatei).....	66
Tabelle 3.11: Zusammenfassung von ermittelten Werten aus den Simulationsdurchläufen mit der Beleuchtung von 11.000 Personen	67
Tabelle 3.12: Vergleich der aus der Simulation ermittelten Daten des Durchschnittskühlschranks.....	70
Tabelle 3.13: Parameter der Kühlschränke (Auszug aus Parameterdatei)	71
Tabelle 3.14: Zusammenfassung von ermittelten Werten aus den Simulationsdurchläufen mit Kühlschränken von 25.000 Personen.....	72
Tabelle 3.15: Vergleich der aus der Simulation ermittelten Daten des Durchschnittskühlschranks.....	73
Tabelle 3.16: Parameter der Gefriergeräte (Auszug aus Parameterdatei)	74
Tabelle 3.17: Zusammenfassung von ermittelten Werten aus den Simulationsdurchläufen mit Gefriergeräten von 34.000 Personen.....	74
Tabelle 3.18: Aufteilung der Programme auf alle Waschvorgänge von Waschmaschinen (Loadcurve_Allocation).....	77

Tabelle 3.19: Parameter für die Waschmaschinen (Auszug aus Parameterdatei).....	78
Tabelle 3.20: Zusammenfassung von ermittelten Werten aus den Simulationsdurchläufen mit Waschmaschinen von 50.000 Personen	79
Tabelle 3.21: Parameter für die Geschirrspüler (Auszug aus Parameterdatei)	81
Tabelle 3.22: Zusammenfassung von ermittelten Werten aus den Simulationsdurchläufen mit Waschmaschinen von 50.000 Personen	82
Tabelle 3.23: Zusammenfassung von ermittelten Werten aus einem Simulationsdurchlauf mit Geräten von 50.000 Personen.....	83
Tabelle 4.1: Aufteilung der Geräte im Sektor Büro	85
Tabelle 4.2: Einstellungen für den DSM-Output der Bürogeräte (Auszug aus Parameterdatei).....	85
Tabelle 4.3: Aufteilung der Geräte im Sektor Audio-Video.....	86
Tabelle 4.4: Einstellungen für den DSM-Output der Bürogeräte (Auszug aus Parameterdatei).....	86
Tabelle 4.5: Einstellungen für den DSM-Output für die Beleuchtung (Auszug aus Parameterdatei)	86
Tabelle 5.1: Gegenüberstellung der wichtigsten Parameter der verschiedenen exemplarischen DSM- Systeme.	98