

DIPLOMARBEIT

Analyse und Benchmarking von Energieverbrauchsdaten

ausgeführt zur Erlangung des akademischen Grades
eines Diplom-Ingenieurs unter der Leitung von

Professor

O. Univ. Prof. Dipl.-Ing. Dr.techn. Dietmar Dietrich

Betreuer

Dipl.-Ing. Friederich Kupzog

am

Institut für Computertechnik (E384)

der Technischen Universität Wien

durch

Klaus Pollhammer BSc.

0126339

Rüdengasse 3/15, 1030 Wien

Wien, Juni 2008

Kurzfassung

Diese Diplomarbeit befasst sich mit dem Thema der Analyse und der mathematischen Aufbereitung von Energieverbrauchsdaten. Im Zuge dieser Arbeit wird eine Möglichkeit vorgestellt, den Energieverbrauch unterschiedlicher Geschäftsstandorte, wie zum Beispiel einzelner Filialen einer Filialkette einzuschätzen und zu bewerten, wofür eine neue Benchmark entwickelt wurde. Zusätzlich wird eine mathematische Methodik vorgestellt, mit Hilfe derer aus einem Lastprofil, also dem Summenstromverbrauch einzelner Stromverbraucher gemessen über dem Messzeitraum eines Tages, berechnet werden kann, ob bestimmte Verbraucher eingeschaltet waren oder nicht. Darüberhinaus wird gezeigt, wie eine Modellierung des Verbrauchsverhaltens von Energieverbrauchern erfolgen kann. Abschließend wird eine beispielhafte Umsetzung der Modelle und Algorithmen in MatLab demonstriert.

Abstract

The topic of this thesis is the analysis and computational preparation of energy-consumption data. For this preparation the development of a new benchmarking algorithm for load profiles is described. The load profile stands here for the sum of the power consumptions of different electrical devices. In addition to that a mathematical method is developed for deciding only from the load profile whether a device was turned on or off. Fundamental for both functions is the modeling of different devices, which is also covered in this thesis. Finally an implementation in MatLab of the described models and algorithms is described.

Danksagung

Besonderer Dank gebührt an dieser Stelle Thomas Frank und allen Mitarbeitern der Envidatec GmbH für die Bereitstellung der Daten und die Unterstützung während der Arbeit an dieser Diplomarbeit.

Ebenso geht ein Dankeschön an Professor Dietmar Dietrich und Friederich Kupzog für die hervorragende Betreuung seitens des Instituts für Computertechnik.

Zu guter Letzt möchte ich mich bei meiner Familie und speziell bei meiner Freundin Elisabeth für die Geduld und den Rückhalt, den sie mir entgegengebracht haben, bedanken.

Ohne sie wäre diese Diplomarbeit nicht in dieser Form möglich gewesen.

Inhaltsverzeichnis

1. Einleitung	1
1.1 Motivation.....	1
1.2 Aufgabe.....	3
1.2.1 Webportal My-JEVis.....	3
1.2.2 Aufgabenstellung.....	5
1.3 Herausforderung	7
2. Benchmarks für Energieversorgung und -verbrauch	8
2.1 Was sind Benchmarks?.....	8
2.2 Benchmarking im Bereich der Stromnetze	10
2.2.1 Bewertungsverfahren für die Versorgungssicherheit von Stromnetzen	10
2.2.2 Bewertung von Kraftwerken am Beispiel von Windkraftanlagen.....	11
2.2.3 Bewertung von Lastfußmodellen am Beispiel des europäischen Stromnetzes.....	12
2.3 Benchmarking im Bereich der Endbenutzer	13
2.3.1 Bewertung des Stromverbrauchs von Privathaushalten	13
2.3.2 Automatische Klassifizierung des Stromverbrauchs einzelner Energiekunden ...	14
2.4 Benchmarking im Bereich der elektrischen Systeme und Geräte	15
2.4.1 Messung des Energieverbrauchs	16
2.4.2 Bewertung des Energieverbrauchs elektronischer Geräte	17
2.4.3 Fehlende Bewertungskriterien für elektronische Geräte	20
2.5 Benchmarking auf Chipebene.....	21
2.5.1 Benchmarking-Methoden für Mikroprozessoren	21
2.5.2 Energiebenchmarking in der Schaltungstechnik	23
2.6 Probleme beim Benchmarking.....	24
2.6.1 Widersprüche beim Begriff „Benchmark“	24
2.6.2 Überblick über die Bewertungsmethoden für den Energieverbrauch.....	26
2.6.3 Notwendige Entwicklungen im Bereich der Energiebenchmarks	28
3. Modellierung von Energieverbrauchern	30
3.1 Bestehende Vorgangsweisen bei der Datenanalyse	30
3.2 Vergleich vorhandener Modelle mit realen Daten.....	32
3.2.1 Modell für eine Kälteanlage	32
3.2.2 Modell für eine Lüftungsanlage	33
3.2.3 Modell für die Beleuchtung.....	34

3.2.4 Modell für die Grundlast	35
3.2.5 Unterschiede zwischen Modell und Realität	36
3.3 Neuentwicklung der Modelle.....	38
3.3.1 Entwicklung eines Modells für Kälteanlagen.....	38
3.3.2 Entwicklung eines Modells für Lüftungsanlagen.....	41
3.3.3 Entwicklung von Modellen für die Beleuchtung.....	42
3.3.4 Entwicklung eines Modells für die Grundlast	43
3.4 Unterschied zwischen neuem und altem Modell	44
3.5 Parametrisierung der Modelle.....	45
4. Bewertung von Energieverbrauchsdaten.....	48
4.1 Vorhandene Ansätze zum Vergleich von Energieverbrauchsdaten	48
4.2 Entwicklung einer aussagekräftigen Benchmark.....	51
4.3 Mögliche Erweiterung der neuen Benchmark.....	53
5. Erkennung einzelner Energieverbraucher	55
5.1 Problematik der Erkennung	55
5.2 Verschiedene Möglichkeiten zur Erkennung.....	56
5.2.1 Fourieranalyse	56
5.2.2 Korrelationskoeffizient zweier Signale	57
5.2.3 Pattern Matching	58
5.2.4 Fuzzy Logic und selbstlernende Algorithmen.....	59
5.3 Lösung zur Erkennung einzelner Energieverbraucher.....	59
5.3.1 Erkennung des Vorhandenseins aller Verbraucher.....	60
5.3.2 Beschreibung der Vorgehensweise zur Detektion von Verbrauchern.....	61
5.3.3 Umsetzung für komplexe Lastprofile.....	63
5.3.4 Mathematische Beschreibung der Vorgehensweise	63
5.4 Verbesserung des Ergebnisses durch Einbeziehung der Korrelation.....	64
5.5 Mögliche Erweiterungen der Detektionsfunktion.....	65
6. Implementierung in MatLab.....	68
6.1 Entwicklung spezieller Datentypen	68
6.1.1 Strukturierte Datentypen in MatLab.....	69
6.1.2 Datentypen für Stromverbraucher und Geschäfte	70
6.2 Implementierung der Modelle.....	71
6.3 Implementierung der Funktion zur Berechnung einer Benchmark.....	72
6.4 Implementierung der Funktion zur Erkennung von Energieverbrauchern.....	72
6.5 Notwendige Zusatzfunktionen	74
Funktion zur Unterscheidung von Arbeitstagen und Sonntagen	74
7. Zusammenfassung & Ausblick	76
Anhang A	83
Specification of JEVIs-MatLab Functions	83
Document-Information	83
Information about parameter names:.....	83

Information about data-types.....	83
Specification of the MatLab/GNU Octave-Functions	84
Information about JEVIS-MatLab Functionalities	86
Document-Information	86
MatLab vs. GNU Octave	86
About the package	86
Anhang B.....	89
funcFilterBenchmark.m	89
funcFilterDetection.m	90
funcGetTimePointIndex.m.....	92
funcIsDaySunday.m.....	92
funcModelFabric.m.....	93
funcNormMeanError.m	96
funcNumberOfPossibleCombinations.m	96
structDevice.m	97
structShop.m	98
Anhang C	99
Werte Filiale 1.....	99
Werte Filiale 2.....	102
Literaturverzeichnis.....	105

1. Einleitung

Im folgenden Kapitel der Diplomarbeit soll zu allererst die Frage behandelt werden, warum in letzter Zeit sowohl im Industriebereich, als auch in den Privathaushalten verstärkt Augenmerk auf energiesparende und energieoptimierende Maßnahmen gelegt wird. Danach wird näher auf die dieser Diplomarbeit zugrunde liegende Aufgabenstellung eingegangen.

1.1 Motivation

In den letzten Jahren ist ein immer stärker werdendes Bedürfnis zur effektiveren Nutzung von Energie zu beobachten. Sowohl in der Industrie, als auch im privaten Bereich wird festgestellt, dass es notwendig ist mit der vorhandenen Energie, sei es nun in Form von Elektrizität, aber auch anderen Energieformen, sparsamer umzugehen.

Ein starker Motor für diese Entwicklung sind natürlich die stark gestiegenen Preise für alle Energieformen, man denke nur an den kontinuierlichen Preisanstieg bei Treibstoff bzw. Rohöl. Aber nicht nur das Drehen an der Preisschraube bestärkt diese Entwicklung, sondern auch die Förderung von Maßnahmen, welche dazu beitragen die Energieeffizienz zu steigern wie zum Beispiel die Passivhaus-Technologie, die Förderung von alternativen Energieformen oder Projekte wie das 10.000-Dächer-Programm, bei dem die Installation von Solarkollektoren finanziell unterstützt wird. Solche Maßnahmen tragen wesentlich dazu bei, dass die Zuständigen von immer mehr Betrieben aber auch Privatpersonen sich Gedanken darüber machen wo und wie auf einfache Art und Weise Energie (und dadurch Geld) gespart werden kann.

Das Umweltbewusstsein steigt kontinuierlich und immer mehr Menschen machen sich Gedanken über ihren so genannten „Ökologischen Fußabdruck“, wobei dieser in einem hohen Maße vom individuellen Strom- und Energieverbrauch abhängig ist.

Besonders, beinahe in jedem Haushalt vorhandene, Stromverbraucher wie Fernseher und Personal Computer entpuppen sich bei Analysen als wahre Energiefresser und sollten nicht im Standby-Betrieb betrieben, sondern immer ganz abgeschaltet werden. Es sind solche oft sehr einfachen Dinge, durch die sich, wenn sie konsequent umgesetzt würden, eine beträchtliche Menge an Geld und Energie einsparen lassen würden. Dadurch erfolgt eine immer stärkere Verankerung im Bewusstsein aller, dass jeder einzelne einen wichtigen Beitrag zum Schutz des Klimas und der Umwelt leisten kann.

Laut einer Studie des deutschen Fraunhofer-Instituts entfielen im Jahr 2001 18% des Gesamtstrombrauchs aller deutschen Haushalte auf so genannte Leerlaufverluste. Ebenso aus dem Endbericht dieser Studie ist folgendes Zitat: „Das Bundesumweltamt hat errechnet, dass durch Leerlaufverluste in den deutschen Haushalten und Büros jährlich Strom im Wert von 3,5 Mrd. Euro verschwendet wird“. [FRA05]

Speziell im Bereich der Industrie aber auch in kleineren und mittelgroßen Unternehmen findet ein Umdenken in Richtung Energiemanagement statt, da die Verantwortlichen den doppelten Nutzen des Energiesparens erkennen. Einerseits kann alleine durch die Reduzierung des Energieverbrauchs sehr viel Geld gespart werden. Selbst die Kosten, welche für eine Optimierung des Energieverbrauchs anfallen, sind auf diese Weise oft in kürzester Zeit wieder eingenommen. Aber nicht nur die möglichen Einsparungen machen solch ein Vorgehen für Betriebe interessant. Auch das gesteigerte Umweltbewusstsein der Konsumenten und Kunden lässt viele Betriebe auf diesen Zug aufspringen, ist doch der Werbewert einer „ökologischen Vorgehensweise“ beispielsweise in der Fertigung durchaus beträchtlich.

Ein gutes Beispiel für solch einen gesteigerten Werbewert durch eine Entwicklung in Richtung einer „ökologische Herstellung“ ist in den letzten Jahren auch im Bereich von Lebensmitteln zu beobachten. Für die so genannten „Bioprodukte“ sind die Konsumenten gewillt mehr zu bezahlen, da mit diesem Attribut eine sehr viel höhere Qualität verbunden wird.

Dieser Trend greift, wie schon erwähnt, nun immer stärker auf klassische Industriebetriebe über, deren Betreiber das positive Image, das eine solch ökologische Vorgehensweise mit sich bringt, als potentiell Argument einsetzen um Kunden anzuwerben, oder sogar von anderen Betrieben abzuwerben, die (noch) keine umweltverträgliche Fertigung vorzuweisen haben.

Dieses Umdenken das Energiebewusstsein betreffend hat dazu geführt, dass immer mehr Betriebe die Dienste so genannter Energiedienstleister in Anspruch nehmen, die ihnen helfen sollen durch maßgeschneiderte Einsparungs- und Optimierungsabschätzungen ein positives Image (und Einsparungen) zu erlangen. Diese Sparte an Unternehmen steht vor der großen Herausforderung, dass sich innerhalb sehr kurzer Zeit, die Menge an potentiellen Kunden vervielfacht hat, da es sich nicht mehr nur für sehr große Betriebe rechnet in diesem Bereich zu investieren, sondern eben auch kleinere Betriebe hier Möglichkeiten sehen. Allerdings ist hier ebenfalls anzumerken, dass nicht alleine der starke Werbewert zu einem Umdenken in diesem Bereich führt, sondern vor allem die potentiellen Einsparungen. Noch sind wir also ein Stück davon entfernt, dass z. B. eine Bäckerei damit wirbt, dass die Ware mit Hilfe besonders energiesparender Geräte gefertigt wird.

Da die Auftragslage für Energiedienstleistungsunternehmen kontinuierlich steigt, werden immer leistungsfähigere Werkzeuge benötigt, welche bei Analyse, Optimierung und Überwachung von Energiedaten unterstützend wirken können. Zu Hilfe bei der Erstellung solcher Werkzeuge kommt den Entwicklern dabei die immer stärkere Vernetzung im Bereich der Telekommunikation, welche das Sammeln und Verteilen von Daten stark vereinfacht. Es muss dadurch keine eigene Infrastruktur für diese Aufgaben geschaffen werden. Die weltweiten Datennetze werden also immer stärker mit einbezogen weil auf diese, relativ günstige, Art und Weise bereits eine Vernetzung erfolgt ist, die bei verschiedensten Arten des Monitorings oder der Überwachung sehr hilfreich sein kann. Um die Weiterentwicklung solch eines Werkzeuges soll es bei dieser Diplomarbeit gehen.

1.2 Aufgabe

Wie schon im vorigen Abschnitt erwähnt handelt es sich beim Thema dieser Diplomarbeit um die Weiterentwicklung eines Instruments zur Überwachung und Analyse von Energieverbrauchsdaten. Konkret geht es dabei um das My-JEVis¹-Webportal, welches von der Envidatec GmbH entwickelt worden ist.

Im ersten Teil des folgenden Unterkapitels soll dieses Webportal vorgestellt werden, da die Aufgabenstellung im Kontext von My-JEVis entstanden ist und es sich dabei um die Weiterentwicklung desselben handeln soll. Die Envidatec GmbH hat auch den Anstoß für diese Diplomarbeit gegeben. Im zweiten Teil des Unterkapitels soll eben jene Aufgabenstellung genau vorgestellt und definiert werden.

1.2.1 Webportal My-JEVis

Beim My-JEVis Webportal handelt sich um eine auf Java-Applets und einer Oracle-Datenbank basierte Webplattform. In der Oracle-Datenbank werden die Messdaten für den Stromverbrauch und die Außentemperaturen von verschiedenen zu beobachteten Objekten gesammelt. Dabei kann es sich um Geschäfte, Tankstellen oder Bürohäuser handeln. Die Messdaten werden durch Messgeräte, welche den Gesamtstromverbrauch einzelner Niederlassungen und Filialen messen, gesammelt und an die Datenbank übermittelt. Das auf Java-Applets basierte Frontend übernimmt dann die Aufgabe diese Messdaten zu visualisieren. Die Struktur des My-JEVis-Webportals ist in Abbildung 1.1 dargestellt.

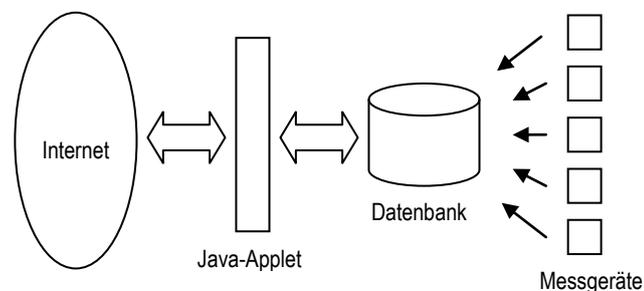


Abbildung 1.1: Struktur des My-JEVis-Webportals

Die momentanen Funktionalitäten der My-JEVis Portals sind alle in der Visualisierungsebene dieser Architektur gebündelt und umfassen die reine Darstellung von Stromverbrauchsmessdaten bzw. Messdaten im Allgemeinen (wie z. B. Temperatur). Bei dieser Darstellung können mehrere Datenreihen in Zeit-Diagrammen gleichzeitig dargestellt werden. Außerdem existieren Funktionen zum Anzeigen eines Rankings verschiedener Filialen, zum automatischen Überwachen der Betriebsführung (mit der Möglichkeit „Alarmmeldungen“ per Fax, Email oder SMS zu generieren) und zu guter

¹ JEVis = Java Envidatec Visualization

Letzt können sich die Benutzer des My-JEVis Portals auch die Rechnungen online über das Portal erstellen lassen. [PAL05]

All diese Funktionalitäten sollen in näherer Zukunft um eine weitere Funktionalität erweitert werden, was im folgenden Abschnitt, welcher die Aufgabenstellung vorstellt, genauer dargelegt werden soll.

Um die immer aufwändigeren Berechnungen und Analysen, die von einzelnen Kunden der Envidatec GmbH gewünscht werden, zu bewältigen wird die bestehende Architektur des My-JEVis-Portals um eine Schnittstelle zu MatLab erweitert (siehe Abbildung 1.2). Dabei soll es möglich sein bestimmte (MatLab)-Funktionen zu festgelegten Zeitpunkten auf bestimmte Datenreihen aus der Datenbank anzuwenden und das Ergebnis der Funktion wieder in die Datenbank einzulesen. Welche Funktionen auf welche Daten angewendet werden sollen, soll dabei vom User individuell über die Weboberfläche bestimmt werden können. Bei der Aufgabenstellung für diese Diplomarbeit handelt es sich nun darum, genau für diese Schnittstelle Funktionalitäten (in MatLab) zu entwickeln, was deswegen in Abbildung 1.2 hervorgehoben wurde.

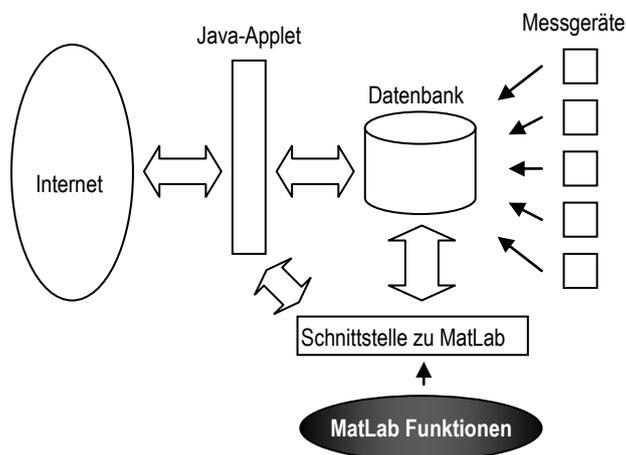


Abbildung 1.2: Erweiterte Struktur des My-JEVis-Webportals

Der Lösungsansatz eine Schnittstelle zu MatLab zu definieren, und dadurch einzelne mathematische Funktionen vom eigentlichen Programm auslagern zu können anstatt beispielsweise in Java zu programmieren, hat den entscheidenden Vorteil, dass solche Funktionen in einfacherer (und übersichtlicherer) Form in MatLab geschrieben werden können. Ein weiterer Vorteil ist, dass MatLab im Gegensatz zu Java darauf optimiert ist Berechnungen größerer Datenmengen effizient und schnell zu bewerkstelligen. Außerdem ist durch solch eine Architektur gewährleistet, dass Veränderungen, Verbesserungen oder Erweiterungen einzelner (mathematischer oder analytischer) Funktionen nicht eine Änderung im Quellcode des Java-Applets nach sich ziehen. Dadurch ergibt sich eine Trennung zwischen den Funktionalitäten und deren Visualisierung, welche auch die Entwicklung dadurch unterstützt, dass verschiedene Entwickler an einer Weiterentwicklung arbeiten können, ohne sich allzu sehr gegenseitig zu behindern. An der Entwicklung dieser Schnittstelle zwischen Datenbank bzw. Java-Applet wird von Seiten der Envidatec GmbH momentan gearbeitet.

1.2.2 Aufgabenstellung

Inhalt dieser Arbeit soll es nun sein, zwei Funktionalitäten für die MatLab-Schnittstelle der MyJEVis-Webplattform zu entwickeln und auch zu implementieren. Im Folgenden soll beschrieben werden worum es sich bei den gewünschten Funktionalitäten handeln soll.

Die erste der beiden Funktionen soll ein Benchmarking der Daten einzelner Filialen durchführen. Diese Funktion soll bewerten ob das Stromverhaltensverhalten innerhalb eines variablen Zeitraums gut oder schlecht war. Dabei ist zu beachten, dass die einzelnen Standorte bzw. Filialen, die untereinander verglichen werden sollen, unterschiedlich groß sind und dadurch der mittlere Stromverbrauch sehr verschieden sein kann. Außerdem soll bei jeder der Filialen auf die dort herrschenden Umweltbedingungen und speziell auf die dort verwendeten Verbraucher eingegangen werden.

Zwar gibt es bereits eine Benchmark-Berechnung, allerdings ist die Aussagekraft dieser Benchmark sehr fragwürdig, handelt es sich doch um eine Benchmark, bei der der Gesamtjahresstromverbrauch mit dem Flächeninhalt normiert wird. Trotz der Normierung lassen sich mit dieser Benchmark vor allem Geschäfte mit ungefähr gleicher Größe und ungefähr gleichem Stromverbrauch vergleichen. Es lassen sich trotz Normierung deswegen vor allem gleich große Geschäfte vergleichen weil natürlich die Menge und Art der Stromverbraucher und auch die klimatischen Bedingungen sehr unterschiedlich sein können, was bei einer Normierung rein auf die Fläche eines Geschäftes nicht berücksichtigt wird. Eine andere Möglichkeit würde z. B. eine Normierung auf den Stromverbrauch darstellen, was allerdings die Verwendung des Gesamtstromverbrauches sinnlos machen würde, könnte man doch schwer den Gesamtstromverbrauch mit dem Gesamtstromverbrauch normieren. Aus diesen Gründen wird es im Zuge dieser Diplomarbeit auch notwendig sein bessere Art des Benchmarkings zu entwickeln und einzuführen.

Bei der zweiten Funktion handelt es sich um eine Erkennungsfunktion, mit der es möglich sein soll aus dem Lastprofil, also den Stromverbrauchsmesswerten, eines Tages zu berechnen ob ein bestimmter Energieverbraucher an diesem Tag eingeschaltet war oder nicht. Teil der Aufgabe ist es dabei herauszufinden, ob solche eine rein mathematische Erkennung überhaupt möglich ist. Notwendig ist diese Funktion deswegen, da gerade bei Geschäftsketten mit sehr vielen Filialen nicht in jeder Filiale Messtechnik zur Überwachung der einzelnen Verbraucher installiert werden kann, da sonst die dadurch entstehenden Kosten mögliche Einsparungen durch Analyse und Optimierung sofort wieder auffressen würden.

Als weitere Bedingung für beide Aufgabenstellungen gilt, dass die oben genannten Funktionen mit möglichst wenig realen Messdaten zu einem möglichst guten Ergebnis kommen sollen. Es wird also davon ausgegangen, dass von einer Niederlassung bzw. Filiale nicht mehr als das 24-Stunden-Lastprofil (4 Messwerte pro Stunde) und die Außentemperatur bekannt sind.

Das alleine kann allerdings nicht ausreichen. Besonders bei der Erkennung einzelner Verbraucher muss zumindest deren Vorhandensein bekannt sein und die größten Betriebs- und Leistungsdaten dieser Verbraucher. Wenn das nicht der Fall wäre könnte man die Entwicklung der zweiten Funktionalität sofort wieder ad acta legen.

Aus diesem Grund kennt man zusätzlich zum Lastprofil und den Messdaten für die Außentemperatur noch die Menge und Art aller Verbraucher einer Filiale und deren Betriebsdaten, also den Soll-

verbrauch dieser Verbraucher. Mit dieser Grundlage an Daten sollen beide Aufgabenstellungen gelöst werden, bzw. abgeschätzt werden ob diese Aufgaben lösbar sind.

Diese, sehr starke Einschränkung bei den zur Verfügung stehenden Daten wurde in der Aufgabenstellung deswegen gewählt, da ebenfalls eines der Ziele dieser Diplomarbeit sein soll herauszufinden ob die Aufgabenstellung auch mit nur äußerst wenigen zu Grunde liegenden Daten lösbar ist und dabei auch zufriedenstellende Ergebnisse zu erzielen sind.

Als Ausgangsdaten zur Entwicklung beider Funktionen stehen reale Messdaten aller Energieverbraucher, des Gesamtstromverbrauchs sowie mehrerer Temperaturfühler einer Kaufhauskette aus den Jahren 2003 bis 2007 zur Verfügung. Bei diesen Messdaten handelt es sich um das Lastprofil der Filialen, die Außentemperatur und Messdaten aller in diesen Filialen vorhandenen Energieverbraucher.

Bevor die Entwicklung der beiden Funktionalitäten begonnen werden kann, soll die vorhandenen Messdaten genau untersucht und analysiert werden. Dieser Punkt ist deswegen zusätzlich in die Aufgabenstellung aufgenommen worden, weil die zu analysierenden Objekte (besagte Filialen einer Kaufhauskette) sehr unterschiedlich in Größe, Stromverbrauch und Kundenvolumen sind und daher sehr unterschiedliche Energieverbrauchsprofile aufweisen. Diese Daten sollen eine Verallgemeinerung der Vorgangsweise ermöglichen.

Außerdem soll durch die Analyse der Daten und der bisherigen Vorgangsweise (auch frühere Analysen, welche per Hand durchgeführt wurden stehen zur Verfügung) ein Gefühl dafür gewonnen werden, worauf es bei einer solchen Analyse ankommt, was gerade bei der Entwicklung einer neuen Benchmark von Bedeutung sein kann.

Wenn die gewünschten Funktionalitäten entwickelt wurden, soll eine Implementierung der Funktionen in MatLab erfolgen.

Zusammenfassend kann man die vorliegende Aufgabenstellung in 3 große Abschnitte teilen, welche in dieser Diplomarbeit in eben dieser Reihenfolge vorgestellt und gelöst werden sollen:

- Analyse- und Modellierungsphase
- Entwicklung der Funktionalitäten
- Implementierung in MatLab

Zusätzlich zu eben diesen Aufgabenpaketen, deren Ausführung ab dem Kapitel 3 behandelt werden soll, wird im Kapitel 2 die Problematik mit Benchmarking-Methoden im Allgemeinen vorgestellt.

Der Zeitrahmen für die gesamte vorgestellte Aufgabenstellung wurde mit 3 Monaten angesetzt, darin waren nicht nur die Analyse und Entwicklung sondern auch die Implementation in MatLab inkludiert.

1.3 Herausforderung

Eine besondere Herausforderung für diese Arbeit stellt sicherlich die sehr geringe Datenbasis dar, schließlich stehen zum Zeitpunkt der Ausführung einer Berechnung nur der Gesamtstromverbrauch, die Außentemperatur und einige Betriebsdaten der Verbraucher einer Filiale zu Verfügung.

Aus diesen wenigen Daten sollen aber möglichst aussagekräftige und gute Ergebnisse erhalten werden, sei es bei der Berechnung der Benchmark, als auch bei der mathematischen Erkennung einzelner Verbraucher.

Eine weitere Herausforderung ist es ein Benchmarking-Verfahren zu entwickeln, das den Vergleich von sehr verschiedenen Objekten ermöglicht und trotzdem eine hohe Aussagekraft behält.

Zu guter Letzt könnte es auch einige Schwierigkeiten bei der Lösung der zweiten Aufgabe geben, wo es gilt eine Berechnung ob ein Verbraucher an einem gewissen Tag verwendet wurde zu entwickeln. Die Herausforderung dabei ist es eine allgemeine mathematische Methodik zu erarbeiten, die eine Lösung dieser Aufgabe darstellt.

2. Benchmarks für Energieversorgung und -verbrauch

Vorab ist zu bemerken, dass Benchmarking bzw. das automatische Bewerten von Objekten und dadurch die Schaffung einer (vereinfachten) Vergleichsbasis von Leistungsmerkmalen ein wichtiges Instrument ist um komplexe Größen und Zusammenhänge auf eine anschauliche Art und Weise darzustellen. Dabei ist natürlich wichtig, dass die vorgenommenen Vereinfachungen gerechtfertigt sind und das Ergebnis nicht verfälschen.

Im Folgenden soll zunächst der Begrifflichkeit „Benchmarking“ und den Anforderungen an eine Benchmark im Allgemeinen diskutiert werden. Hier wird darauf eingegangen wofür man Benchmarks überhaupt braucht und wo sie überall zu finden sind. Außerdem wird gezeigt, dass nicht nur im Bereich der Technik, sondern beinahe in allen Bereichen des täglichen Lebens auf die Aussagekraft von Vergleichszahlen vertraut wird. Dann soll den unterschiedlichen Formen von Benchmarks Rechnung getragen werden, die man im Bereich der Energieversorgung finden kann. Dabei wird das weite Feld der Energieversorgung in 4 verschiedene, von einander unabhängige Bereiche aufgeteilt. Für jeden dieser Bereiche soll dann gezeigt werden, inwieweit Benchmarks bereits realisiert sind. Bei den Bereichen handelt es sich um die Ebene der Energieversorgung und -verteilung repräsentiert durch die Stromnetze, um die Ebene der Energiekunden und Endverbraucher, um die Geräte- und Systemebene und zu guter Letzt um die Ebene der Computerchips und integrierten Schaltkreise.

Abgeschlossen soll das Kapitel werden durch eine zusammenfassende Diskussion, welche Probleme durch Benchmarking auftreten können, beziehungsweise welche Verbesserungen im Bereich des Benchmarkings unbedingt notwendig wären.

2.1 Was sind Benchmarks?

Wie bereits in der Einführung erwähnt soll im folgenden Unterkapitel darauf eingegangen werden, was man im Allgemeinen unter einer Benchmark versteht und warum diese Art der Bewertung in der letzten Zeit immer beliebter wird und der Bewertung durch Benchmarks immer mehr Bedeutung zugemessen wird.

Grundsätzlich lässt sich sagen, dass es sich bei einer Benchmark um eine Vergleichszahl handelt mit Hilfe derer zwei Objekte derselben Art oder Klasse einem Vergleich unterzogen werden können. Wichtig dabei ist nur, dass es sich bei einer Benchmark um eine messbare und dadurch auch vergleichbare Größe handeln. Beim Benchmarking werden komplexe und schwer überschaubare Eigenschaften und Zusammenhänge zu relativ einfachen Vergleichszahlen zusammengefasst, was einen

schnellen Vergleich ermöglicht. Dabei soll hier nicht darauf eingegangen werden, dass der Begriff Benchmark an sich sehr missverständlich ist. Diese Facette des Problems soll in Kapitel 2.6 näher beleuchtet werden, während hier ganz allgemein erklärt werden soll was überhaupt unter Benchmarking verstanden wird.

Der Begriff „Benchmark“ steht in starkem Zusammenhang mit dem Vergleich technischer Daten, insbesondere um Leistungsdaten. Als Beispiel kann hier die Werbung für Kraftfahrzeuge angeführt werden, wo längst nicht nur die Motorleistung, die Drehzahl oder der Treibstoffverbrauch von Bedeutung sind. Auch die Anzahl der Sitze, die Anzahl der Türen, das Kofferraumvolumen oder die Anzahl der Airbags können zwischen verschiedenen Automodellen als Vergleichskriterien herangezogen werden und für einen Autokauf entscheidend sein. Einzig Eigenschaften die schwer messbar sind wie die Farbe der Lackierung eignen sich wenig bis überhaupt nicht für solche Vergleiche.

Gerade beim eben vorgestellten Beispiel ist gut zu sehen, dass das Heranziehen einer einzigen Benchmark nur sehr eingeschränkte Aussagekraft hat und es deswegen zwar mehr oder weniger aussagekräftige Benchmarks gibt, eine einzige dieser Benchmarks wohl immer zu wenig sein wird um ein Objekt zu bewerten. Besonders um das gesamte Verhalten oder die gesamten Eigenschaften eines Objektes abzubilden, ist es nicht ausreichend nur eine einzige Benchmark als Argument anzuführen.

Ein gutes Beispiel dafür, dass es sehr aussagekräftige Benchmarks gibt, die allerdings auch nie alleine entscheidend sein sollten, ist der Quadratmeterpreis beim Wohnungskauf. Dass es sich auch dabei um eine Benchmark handelt ist nicht immer klar. Allerdings ist die Aussagekraft dieser Vergleichszahl durchaus akzeptabel, so kann alleine aus dieser Vergleichsgröße sehr viel über die möglichen Eigenschaften einer Wohnung abgeleitet werden, beispielsweise in welcher Gegend sie liegt oder wie die allgemeine Bausubstanz ist. Natürlich lassen sich solche Dinge nicht von vornherein aus dem Quadratmeterpreis herauslesen. Allerdings ist mit einer gewissen Erfahrung und dem Wissen wo genau die dazugehörige Wohnung liegt sehr gut einzuschätzen ob sich eine nähere Begutachtung auszahlen könnte oder nicht.

Wie man an den Beispielen sehen kann ist einiges an Erfahrung nötig, will man alleine auf Grund einer einzigen Benchmark abschätzen wie gut oder schlecht ein Objekt einzustufen ist. Die zu wenig vorhandene Erfahrung ist es auch, die sehr oft in der Werbung ausgenützt wird, wo die herausragenden Benchmarks zu werbezwecken sehr plakativ angepriesen werden und die weniger herausragenden Vergleichswerte einfach unter den Tisch gekehrt werden.

Im Bereich der Technik sind Benchmarks in der letzten Zeit ein sehr wichtiges Instrument geworden um die Eigenschaften verschiedener, aber vergleichbarer, Objekte untereinander zu vergleichen. Sie finden in sämtlichen Bereichen der Geschäfts- und Industriewelt, wie auch in der Forschung Verwendung. Dabei helfen sie komplexe und meist schwer überschaubare Daten anschaulich darzustellen und die Zusammenhänge so sehr zu vereinfachen, dass eine schnelle Bewertung ermöglicht wird.

Dabei muss klar herausgestrichen werden, dass solch eine Benchmark niemals eine genaue Analyse oder einen Vergleich ersetzen kann. Eine Benchmark kann immer nur einen ersten Eindruck oder eine erste Klassifizierung der vorliegenden Daten bieten und dafür sorgen, dass jene Daten, bei welchen eine genauere Analyse zu interessanten Ergebnissen führen könnte schnell gefunden werden.

Man denke hier nur an den Mittelwert von Messdaten. Jene Datenreihen, welche ungefähr denselben Mittelwert aufweisen sind sehr viel weniger interessant, als die so genannten „Ausreißer“. Diese zu untersuchen kann sehr viel aufschlussreicher sein, als „normale“ Daten zu untersuchen. Der Mittelwert ist hier also auch als eine Art Benchmark aufzufassen, die hilft jene Datenreihen zu finden bei denen eine Untersuchung zu interessanten Ergebnissen führen kann. Der Mittelwert hilft hier also die aussagekräftigen Datenreihen auf effiziente Art und Weise von den weniger aussagekräftigen Datenreihen zu trennen.

Meist werden Benchmarks angewandt, wenn es darum geht zu zeigen, dass durch ein neues Produkt eine bessere Performance erzielt werden kann, als mit einem vergleichbaren Produkt der älteren Generation, oder, was noch öfter der Fall ist, eines Konkurrenzunternehmens. Einige dieser Benchmarks haben sich als Leistungsmerkmale etabliert, man denke nur an die Taktfrequenz eines Prozessors, die ein sehr wichtiges Kaufargument beim Kauf von Personal Computern darstellt. Aber auch in der Wissenschaft und Forschung haben sich Benchmarks etabliert um die Effizienz oder Überlegenheit von neuentwickelten Verfahren zu untermauern.

In der Literatur findet man Benchmarks für beinahe jedes Feld der wissenschaftlichen Forschung. In [WEI90] werden beispielsweise verschiedenste Benchmarks für die Geschwindigkeit von Prozessoren vorgestellt, während in [SIE93] eine Benchmark für die Robustheit eines Systems beschrieben wird. Vor allem im Bereich der IT sind Benchmarks sehr weit verbreitet, da hier Zusammenhänge und Betriebsdaten vorherrschen, welche sonst sehr schwer zu vergleichen und zu interpretieren wären, vergleiche hierfür [MIY88] bzw. [LEV05] Weitere Benchmarking-Methoden, vor allem solche die in den verschiedenen Bereichen der Energieversorgung und des Energieverbrauchs verwendet werden, werden in den folgenden Abschnitten genauer beschrieben und untersucht.

2.2 Benchmarking im Bereich der Stromnetze

Neben dem ausreichenden Vorhandensein von elektrischer Energie ist die permanente und gesicherte Bereitstellung derselben in den letzten Jahren zu jenem Bereich geworden, welcher bei den Energieversorgern immer stärkere Beachtung findet und deren Verbesserung immer wichtiger wird.

2.2.1 Bewertungsverfahren für die Versorgungssicherheit von Stromnetzen

Durch die starke Verbreitung computerunterstützter Technologien und dem Trend zu immer stärkerer Integration solcher Technologien auf beinahe allen Feldern der Industrie, der Verwaltung aber auch im privaten Bereich wird der Fokus auf eine gesicherte und permanente Versorgung mit elektrischer Energie nur noch weiter verstärkt. Diese Verbesserung der Versorgungssicherheit ist deswegen so wichtig, weil die neuen Technologien sehr stark abhängig vom elektrischen Strom sind. So können bereits kurze Stromausfälle ein gewaltiges Chaos in Computersystemen verursachen, bei denen nicht Vorsorge durch die Integration von Stützbatterien oder Notfall-Prozeduren getroffen wurde.

Aus den eben genannten Gründen wird klar, dass eben diese Versorgungssicherheit eine Größe ist, welche für Energieversorgungsunternehmen sehr interessant sein muss, hängt von ihr doch das

Überleben der Energieversorger ab. Das Problem dabei ist auf welche Art und Weise solch eine Größe messbar gemacht werden kann. Die Einschätzung des Energieverbrauchs der Kunden ist für die Energieverbraucher sehr viel weniger interessant als die Einschätzung der eigenen Leistungsfähigkeit, es wird angenommen, dass diese einfach immer Strom brauchen – nur was passiert wenn es einmal nicht so ist?

In [CHO00] werden, ausgehend von [IEEE96], vier verschiedene so genannte Performance-Indizes definiert, bei denen es sich eindeutig um Benchmarks handelt. Mit diesen können die Zuverlässigkeit und Performance von Energieversorgungsnetzen vergleichbar gemacht werden, weshalb sie sich für Energieversorgungsunternehmen besonders gut eignen.

Hier sollen nun die vier von den Autoren der beiden erwähnten Publikationen definierten Benchmarks vorgestellt und betrachtet werden, wobei für die genaue Definition der Benchmarks auf die genannten Quellenangaben verwiesen werden soll:

- System Average Interruption Frequency Index (SAIFI)
- System Average Interruption Duration Index (SAIDI)
- System Average Restoration Index (SARI)
- Delivery Point Reliability Index (DPRI)

Alle hier aufgezählten Benchmarks stehen dabei für einen Quotienten, welcher einen Beobachtungsfaktor mit dem Beobachtungszeitraum normiert, sei es die mittlere Anzahl der Serviceunterbrechungen im Beobachtungszeitraum (SAIFI) oder sei es die durchschnittliche Zeit in der kein Strom bereitgestellt werden kann (SAIDI). Der Beobachtungspunkt liegt dabei immer an der Stelle an der die Energie vom Übertragungsnetz an den Endnutzer übertragen wird.

Zusätzlich zu den oben wird in [CHO00] darauf eingegangen, dass die hier vorgestellten Arten des Benchmarkings, aus dem Grund weil sie auf das gesamte System angewandt werden, dazu tendieren besonders gute und schlechte Werte zu maskieren. Darüber hinaus wird gezeigt, inwieweit man diese Benchmarks verwenden kann um die Energieversorgung zu beeinflussen und welche Referenzwerte für diese Benchmarks angenommen werden können.

2.2.2 Bewertung von Kraftwerken am Beispiel von Windkraftanlagen

Ebenso wie die bereits vorgestellten Benchmarks, deren Hauptaugenmerk darin liegt die Versorgungssicherheit aus der Sicht der Endbenutzer vergleichbar zu machen, können natürlich auch Benchmarks definiert werden, mit deren Hilfe die Güte und Produktionsmenge von Kraftwerken zur Erzeugung elektrischer Energie untereinander verglichen werden können. Diese sind vor allem für Energieversorgungsunternehmen wichtige Vergleichswerte, kann auf deren Grundlage doch auch eine Abschätzung getroffen werden, ob die Versorgung gesichert ist. Durch das Heranziehen solcher Benchmarks können zum Beispiel auch zukünftige Projekte und deren Eigenschaften auf einfache Art und Weise prognostiziert werden.

In [MEI05] wird eine Benchmark vorgestellt, mit deren Hilfe ein Benchmarking verschiedener Topologien für Windfarmen verglichen werden kann. Das kann bei der Planung neuer Windfarmen eine entscheidende Rolle spielen, da dadurch die Rentabilität eines solchen Projektes gut vorhergesagt werden kann. Das Verfahren basiert auf dem Wert der jährlichen Energieproduktion einer

Windfarm (Annual Energy Production *AEP*), wobei auch Erfahrungswerte bezüglich mittlerer Windgeschwindigkeiten mit in die Berechnung einbezogen werden müssen. Die *AEP* berechnet sich dabei wie folgt:

$$AEP = N_h \sum_{i=1}^N \left\{ [F(V_i) - F(V_{i-1})] * \left(\frac{P_{i-1} + P_i}{2} \right) \right\}$$

In dieser Formel stellen N_h die Anzahl der Stunden in einem Jahr, V_i und P_i die normalisierten und durchschnittlichen Windgeschwindigkeiten dar. Bei der Funktion $F(x)$ handelt es sich um die Rayleigh-Wahrscheinlichkeitsfunktion mit deren Hilfe die Verteilung von Mittelwerten der Windgeschwindigkeit beschrieben wird.

Des Weiteren wird an Hand verschiedener Kriterien (Übertragungsdistanz, Windgeschwindigkeit) gezeigt wie dieser Wert als Benchmark für die Bewertung von drei verschiedenen Topologien für Windfarmen verwendet werden kann. In den folgenden Kapiteln wird gezeigt werden, dass viele Vergleichszahlen und Benchmarks sich aus messbaren physikalischen Eigenschaften der zu vergleichenden Objekte ableiten lassen.

2.2.3 Bewertung von Lastfußmodellen am Beispiel des europäischen Stromnetzes

Um den Bereich der Benchmarks in Energieversorgungsnetzen abzuschließen soll zu guter Letzt noch näher auf [ZHO05] eingegangen werden und die dort behandelten Ansätze vorgestellt werden. Hier zeigen die Autoren eine Möglichkeit mit deren Hilfe man das europäische Stromnetz dergestalt abbilden kann, dass die Beobachtung und Bewertung der bisher wenig untersuchten Effekte des grenzüberschreitenden Handels mit elektrischer Energie zulässt. Laut den Autoren werden solche Beobachtungen immer wichtiger, da durch die Liberalisierung des Energiemarktes die so genannten „grenzüberschreitenden“ Energietransporte immer mehr zunehmen und deswegen ein System entwickelt werden muss, mit Hilfe dessen sich solche Transporte bewerten und behandeln lassen.

Auf Grund eines fehlenden Lastflussmodells für ganz Europa haben die Autoren der oben genannten Studie ein solches, vereinfachtes Modell entwickelt und es gleichzeitig verwendet um ebenfalls von ihnen entwickelte Ideen zum Umgang mit Überlast zu testen. Dabei sind diese Ideen für die vorliegende Arbeit weniger von Interesse als das bereits erwähnte Modell. Dieses Modell soll nämlich dazu dienen um die Wirksamkeit verschiedener Methoden zu bewerten, also zu benchmarken. Dazu werden die Methoden auf das Modell angewendet und die Performance der Methoden von externen Bewertungsfunktionen oder Beobachtern bewertet.

Bei dem Modell handelt sich allerdings nicht um ein detailgetreues Modell, denn ein solches ist laut den Autoren nicht realisierbar, da die einzelnen Energieversorger aus datenschutztechnischen Gründen keinen Zugriff auf die detaillierten Lastflussmodelle zulassen. Deswegen wurden an Hand von statistischen Werten wie Bevölkerung und dem bekannten mittleren Stromverbrauch pro Person Black-Box-artige Modelle für das energietechnische Verhalten und die Reaktion auf Lastflüsse der einzelnen Länder Europas entwickelt. Diese Modelle gehen davon aus, dass sich die einzelnen Energienetze auf einen Lastfluss immer auf dieselbe Art und Weise verhalten. Aus diesen Black-Box-Modellen wurde dann ein Modell für das gesamte „System“ der europäischen Lastflüsse zusammen-

gesetzt. Wobei eben nicht die Teilnetze an sich modelliert werden, sondern das Verhalten der lokalen Stromversorgungsnetze auf die Lastflüsse und was oft noch wichtiger ist deren Änderung.

Mit einem solchen Modell ist es nun mögliche verschiedene Arten des Lastmanagements zu benchmarken indem man diese auf das Modell anwendet und das Ergebnis bewertet. Dieser Ansatz ist deswegen bemerkenswert, da hier auf einer sehr globalen Ebene ein Verfahren des Benchmarkings Anwendung findet, das auch in ganz anderen Bereichen verwendet wird. So wird auch zum Beispiel bei vielen Computerbenchmarks (Taktrate, Operationen pro Sekunde oder ähnlichen Werten) ein Beispielprogramm oder Modell als Referenz herangezogen an Hand dessen dann eingeschätzt werden kann wie gut oder schlecht die Performance des zu untersuchenden Objektes sei es nun Prozessor, Grafikkarte oder andere Komponente. Für Informationen und der genaueren Betrachtung solcher Benchmarks sei auf die Kapitel 2.4 und 2.5 verwiesen, welche sich mit Systemen und Systemkomponenten und den darauf angewandten Benchmarking-Methoden beschäftigen.

2.3 Benchmarking im Bereich der Endbenutzer

Nachdem das vorige Unterkapitel vor allem der Verteilung und Produktion von elektrischer Energie und den in diesem Bereich üblichen Benchmarks gewidmet war, soll nun die Seite der Endbenutzer beleuchtet werden. Damit sind alle Kunden der Energieversorger gemeint, seien es nun Betriebe oder Privathaushalte. Dabei geht es allerdings weniger um das Vorhandensein der elektrischen Energie bei den einzelnen Kunden, da dieser Part schon einige Absätze vorher beleuchtet wurde, sondern es sollen Bewertungsmöglichkeiten des Energieverbrauchs der einzelnen Endbenutzer untersucht werden.

2.3.1 Bewertung des Stromverbrauchs von Privathaushalten

Meist wird als Benchmark im Bereich der Endbenutzer und Endkunden der Energieversorgungsbetriebe der Gesamtstromverbrauch über einen gewissen definierten Zeitraum herangezogen. Als Vergleichsbasis ist dieser auch sicherlich ausreichend, doch hat eine Benchmark meist auch den Anspruch, dass dadurch eine wirklich Bewertung erfolgt. Daran mangelt es sicherlich bei alleiniger Verwendung des Gesamtstromverbrauches zur Bewertung des Stromverhaltens eines einzelnen Endbenutzers. Ob viel oder wenig Strom verbraucht wurde hängt nämlich längst nicht nur vom Verhalten des Kunden ab, sondern auch von nicht beeinflussbaren Variablen. So sollten immer auch die näheren Umgebungsvariablen in solche eine Bewertung mit einbezogen werden. Zum Beispiel ist der Energieverbrauch auch immer auf direkte oder indirekte Art und Weise von klimatischen Bedingungen wie der Temperatur abhängig. Ebenso wie diese Variablen gibt es aber noch einige weitere Einflussfaktoren auf den Stromverbrauch wie die Qualität der installierten Geräte, die Anzahl der im Haushalt wohnenden Personen und einige mehr.

Damit jeder Privatkunde den eigenen Stromverbrauch einschätzen lernt und eine Bewertung desselben durchführen kann, hat das österreichische Energieversorgungsunternehmen „Wien Energie“ eine Internetseite entwickelt, auf der jeder, nach Eingabe einiger Daten, eine Bewertung des eigenen Stromverbrauchs erhält. [ESC07] Bemerkenswert ist hier, dass diese Bewertung abhängig von den genauen Daten der Energieverbraucher durchgeführt wird. Diese Daten enthalten die Bausubstanz,

genauso wie das Wasserverbrauchsverhalten und das Heizungssystem. Alle diese Faktoren werden in die Bewertung mit einbezogen. Abhängig von all diesen Kriterien wird der individuelle Energieverbrauch in Bezug gesetzt zum „idealen“ Konsumenten, der zwar dieselben Geräte installiert hat, diese Geräte allerdings auf ideale Art und Weise verwendet.

Dieser Ansatz ist sehr interessant, weil dadurch erstmals ein Werkzeug für Privatkunden zur Verfügung gestellt wird, mit Hilfe dessen der Energieverbrauch eingeschätzt werden kann. Zusätzlich zur Bewertung werden aber auch Einsparungsmöglichkeiten und Optimierungspotentiale aufgezeigt. Hat der Kunde zum Beispiel angegeben, dass es sich beim Herd um ein „altes“ Modell mit Gusseisen-Herdplatten handelt, könnte einer der Vorschläge bei der Auswertung sein, dass der Kunde auf einen modernen Herd mit Ceran- oder Induktionskochplatte umsteigen sollte. In Abbildung 2.1 kann man erkennen, wie das Ergebnis einer solchen mit Hilfe des Energiesparchecks durchgeführte Bewertung aussehen könnte. Der Screenshot enthält nur den Überblick über die Bewertung, die gesamte Auswertung ist sehr viel detaillierter ausgeführt. [ESC07]

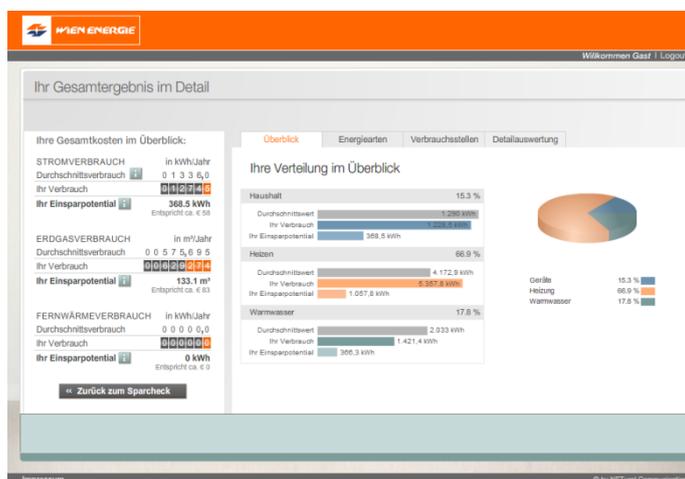


Abbildung 2.1: Teil einer Energieverbrauchsbewertung durch den Wien-Energie-Energiesparcheck

2.3.2 Automatische Klassifizierung des Stromverbrauchs einzelner Energiekunden

Wie man bei aus vorigen Ausführungen erkennen kann sind für eine Bewertung des Energieverhaltens umfassende Kenntnisse aller möglichen Einflussgrößen notwendig. Genau das kann zu großen Problemen bei Betrieben und Unternehmen mit mehreren Filialen führen. Selbst wenn ein vergleichbares Werkzeug wie der in den Absätzen weiter oben beschriebene Energiecheck vorhanden wäre, müssten die notwendigen Daten zuerst gesammelt und danach dem System bekannt gemacht werden.

Ein anderes Problem sowohl für den Bereich der Privatkunden als auch für den Bereich der Unternehmen ist, dass es momentan noch keine wirkliche Nachfrage nach Benchmarking-Methoden für den Energieverbrauch gibt. Es gib zwar einen leichten, bemerkbaren Trend hin zu mehr energieschonenden Vorgehensweisen das wichtigste Werkzeug für die Bewertung des Energieverbrauchs fehlt allerdings. Oft wird geraten als „Energiekennzahl“ den Gesamtstromverbrauch eines Jahres

normiert mit der Gesamtfläche des Geschäftes oder der Wohnung als Vergleichszahl zu nehmen. Dass diese Vergleichszahl allerdings in Wirklichkeit nur Vergleiche mit dem Stromverbrauch desselben Geschäftes nur unterschiedlicher Jahre zulässt, wird meistens verschwiegen. Wie bereits erwähnt wurde sind unterschiedliche Geschäfte, selbst wenn sie die gleiche Größe haben, auf Grund der unterschiedlichen Geräte, die in diesen Geschäften installiert sind wenig bis gar nicht zu vergleichen.

Eine Lösung dieses Problems könnte so aussehen, dass die einzelnen Energiekunden (egal ob nun Unternehmen oder Privatkunden) mittels mathematischer Methoden zu klassifiziert werden. Bei [LO04] ist ein Ansatz dargelegt, wie solch eine mathematische Klassifizierung aussehen könnte. Der vorgestellte Ansatz basiert auf neuronalen Netzen. Dabei werden die verschiedenen Lastprofile automatisch von einem mathematischen Algorithmus verschiedenen Clustern zugeteilt. Als Cluster gelten hier verschiedene Klassen von Energiekunden wie „Privatkunde“, „Geschäft“ und so weiter.

Laut den Autoren funktioniert diese Technik den Umständen entsprechend sehr gut, allerdings relativieren sie selbst ihre Ergebnisse, steht doch wenige Zeilen darunter: „If two input vectors are very similar, the competitive layer probably will put them in the same class.“ Hier wird einer der größten Kritikpunkte an dieser Methode der Klassifizierung also wieder nicht entkräftet, denn wenn sich zwei Lastprofile sehr ähnlich sind (auch wenn sie von unterschiedlichen Verbrauchertypen stammen) werden sie in dieselbe Klasse oder dasselbe Cluster eingeordnet, wodurch dann wieder eigentlich unterschiedliche Charakteristiken nicht unterschieden worden sind. Darüberhinaus gehen die Autoren davon aus, dass alle auftretenden Lastprofile immer von verschiedenen Verbrauchertypen stammen und es wird nicht darauf eingegangen was passiert, wenn es viele ähnliche Verbrauchercharakteristiken gibt.

Abschließend ist zu sagen, dass im Bereich der Endkunden sehr viel Potential für neue Benchmarking Methoden besteht. So gibt es zwar Ansätze für die Bewertung und Optimierung des Energieverbrauchs von Privatkunden, doch bei Kunden aus dem Bereich der Unternehmen klafft hier eine eklatante Lücke. Was fehlt ist ein wirkliches Konzept was mit solch einer Bewertung ausgesagt werden soll, beziehungsweise wie eine solche Bewertung überhaupt durchgeführt werden könnte. Alleine nur den gesamten Strom- oder Energieverbrauch im Allgemeinen zu messen und zu sagen, dass die Messwerte, welche den niedrigeren Stromverbrauch aufweisen besser, zu bewerten sind, kann nicht ausreichend sein. Der individuelle Energieverbrauch ist von zu vielen Faktoren abhängig, wovon die Bausubstanz der Häuser und die klimatischen Bedingungen noch die augenscheinlichsten sind, deswegen kann die Höhe des Gesamtstromverbrauchs immer nur einer von mehreren Indikatoren für einen „guten“ Stromverbrauch sein, aber niemals das allgemeingültige Element einer Bewertung.

2.4 Benchmarking im Bereich der elektrischen Systeme und Geräte

In diesem Abschnitt sollen Benchmarks thematisiert werden, mit denen es möglich ist, den Energieverbrauch einzelner elektrischer Systeme und Geräte zu bewerten. Solche Benchmarks sind deswegen interessant, da genau im Bereich der elektrischen Geräte wahre „Energiefresser“ zu finden sind

und eine genauere Untersuchung des Energieverbrauchs einzelner Geräte sicherlich einige Beachtung geschenkt werden sollte.

2.4.1 Messung des Energieverbrauchs

Die erste Energie-Benchmark, die einem im Bezug auf elektrische Geräte einfällt ist natürlich der mittlere Energieverbrauch, meist gemessen in Kilowattstunden (kWh). Bei einer Wattsekunde handelt es sich um die physikalische Maßeinheit der zeitbezogenen Arbeit, auch Leistung genannt. Die physikalische Einheit der Arbeit heißt Joule, bei der zeitbezogenen Arbeit handelt es sich also um einen Mittelwert. Die Schreibweisen kW/h bzw. kW sind nicht äquivalent zur kWh und deswegen falsch im Zusammenhang mit Energieverbrauch. Die Wattsekunde ist in der Fachliteratur eindeutig definiert und zwar als „Die Einheit der Leistung, Joule pro Sekunde, nennt man auch Watt“. [PRE94 S.37] Physikalisch ist die Wattsekunde damit äquivalent mit der SI-Einheit Joule, eine Wattstunde würde 3600 Joule entsprechen.

Um nun einzelne Geräte an Hand dieser Benchmark zu klassifizieren reicht es meist aus, auf die allgemeinen Betriebsparameter oder das Referenzblatt zu sehen, da dort der mittlere Energieverbrauch angegeben sein sollte. Alleine an Hand dieser Einteilung können die größten Energieverbraucher sehr einfach gefunden werden. Was allerdings überraschen ist, ist dass sich unter diesen nicht nur die offensichtlichen Verdächtigen wie Elektroherd und Waschmaschine finden. Denn immer öfter wird der Hauptanteil an elektrischer Energie von Computern, Kommunikations- und Unterhaltungsgeräten verbraucht, besonders Plasma-Fernseher stechen als Negativbeispiele heraus.

Tabelle 1: Mittlere Jahresenergieverbrauch einiger Geräte (Werte für das Jahr 2004) [FRA05]

Gerät	Energieverbrauch (kWh/a)
Herd	360
Waschmaschine	143,5
Personal Computer	70
Fernseher (Kathodenstrahl)	183,9
Fernseher (Plasma)	171,3
Fernseher (LCD)	619,7
Videobeamer	400,1
DVD-Recorder	66,9

In einer Studie des deutschen Fraunhofer-Institutes über die Problematik, welche durch den Leerlaufverbrauch von Haushalts- und Bürogeräten entsteht, wird angegeben, dass im Jahr 2004 in allen deutschen Haushalten 10932 GWh alleine von Geräten aus dem Informations- und Kommunikationsbereich wie Computer, Fernseher und Spielkonsolen verbraucht wurden, während der Energieverbrauch von klassischen Haushaltsgeräten wie Elektroherden und Waschmaschinen bei dagegen

bescheidenen 2101 GWh lag. Diese Zahlen sind dadurch begründet, dass die Geräte erstens sehr viel Strom verbrauchen und zweitens sehr viel längere Einsatzzeiten bei diesen Geräten auftreten als bei den klassischen Verbrauchern. Außerdem tritt bei Geräten aus der Klasse der Unterhaltungselektronik ein enormer Stand-By-Verbrauch auf, so dass ein Herd zwar eine größere Leistungsaufnahme hat, allerdings niemals so lange verwendet wird wie ein Fernseher oder Personal-Computer. Der mittlere Jahresverbrauch einiger Geräte ist in Tabelle 1 (siehe vorige Seite) angeführt, wobei dabei auch der Stand-By-Verbrauch berücksichtigt wurde. [FRA05]

Wie man aus den Angaben zum Gesamtverbrauch der einzelnen Gerätegruppen entnehmen kann, welche bereits im vorigen Absatz angeführt wurden, hat die Computer- beziehungsweise Kommunikationstechnik innerhalb von relativ kurzer Zeit das „Energieverbrauchsmonopol“ in Haushalten und Büros übernommen. Diese Entwicklung ist deswegen erst relativ kurz Zeit zu beobachten, weil Personal Computer erst seit den 1980er Jahren weite Verbreitung gefunden haben. Eine Optimierung des Energieverbrauches sollte also genau im Bereich solcher Geräte stattfinden, da hier (alleine durch das größere Verbrauchsvolumen) naturgemäß höhere Einsparungen möglich sind. Was für eine solche Optimierung allerdings fehlt sind die Bewertungskriterien, abgesehen vom durchaus nicht zu unterschätzenden Kriterium „geringer Gesamtstromverbrauch“.

Ähnlich der Problematik, dass es wenige bis gar keine Benchmarks zum Bewerten des Energieverbrauches von Mikrochips und anderen technischen Schaltungen gibt (siehe dafür Kapitel 2.5), verhält es sich auch hier. Meist finden sich so genannte Richtwerte für den Stromverbrauch von Haushaltsgeräten. So gibt es in der Europäischen Union zwar eine gesetzlich vorgeschriebene Kennzeichnung des Energieverbrauches von Haushaltsgeräten, das so genannte „EnergieLabel“. Dieses EnergieLabel ist allerdings nur für Haushaltsgeräte und nicht für elektronische Geräte vorgeschrieben, weil diese nicht per se als Haushaltsgeräte gelten. [EL08]

2.4.2 Bewertung des Energieverbrauchs elektronischer Geräte

Was bei Herd, Waschmaschine und Mikrowelle ein ausreichendes Bewertungskriterium für den Energieverbrauch ist, da diese Verbraucher im Betrieb weitgehend konstanten Energieverbrauch aufweisen und ansonsten beinahe keine Leistungsaufnahme zeigen, stellt sich bei Computern und beispielsweise Flachbildfernsehern als ein nicht ausreichendes Kriterium dar. Diese Geräte verbrauchen, auch wenn sie nicht eingeschaltet sind, beträchtliche Mengen an Energie, nicht alleine durch den, bereits höchst umstrittenen, Stand-By-Modus, sondern auch deswegen weil in vielen Netzteilen die Versorgungsspannung zuerst unter hohen Verlusten transformiert werden muss. Eine ähnliche Problematik ergibt sich bei den Ladegeräten mobiler Unterhaltungs- und Telekommunikationselektronik (MP3-Player, Mobiltelefon), welche oft nach der Benutzung einfach nicht von der Steckdose entfernt werden und dadurch große Mengen Energie verschwendet wird. Diese und noch einige andere Faktoren müssten in eine Bewertung solcher Geräte mit einfließen, was eine solche zugegebenermaßen sehr komplex machen könnte, aber unmöglich durchzuführen wäre es sicherlich nicht. Wie bereits im Abschnitt über die Bewertung des Energieverbrauchsverhalten von Endkunden (Kapitel 2.3) täuscht hier eine vorhandene Möglichkeit (Berechnung des Gesamtstromverbrauches) über die Lücken hinweg, welche durch offensichtliche Vernachlässigung einiger wichtiger Einflussfaktoren entsteht.

So finden sich zwar etliche Benchmarks, welche auf die Performance und Leistungsfähigkeit der zu untersuchenden Objekte eingehen, wie zum Beispiel einer Arbeit von Markus Levy aus dem Jahr 2005, in welcher genau beschrieben wird, wie man die Leistungsfähigkeit für Audio- und Videogeräte (in diesem Fall DVD-Player, MP3-Player und ähnliches) bewerten kann. Ein Statement des Autors sticht beim Lesen der Veröffentlichung ins Auge, so räumt er am Anfang des zweiten Abschnittes des Artikels ein, dass das allerbeste Benchmarking-Programm für ein Gerät grundsätzlich jene Applikation ist, für die das Gerät verwendet werden soll, also die Bewertung immer an Hand des Verwendungszwecks erfolgen soll. Wörtlich ist in dem Artikel zu lesen: „The best digital entertainment benchmark is the exact application that will be run on the device.“ [LEV05]

Damit wird klar, dass die Leistungsfähigkeit eines DVD-Players am besten an Hand der Performance bewerten kann, welche bei der Belastung mit einer typischen Applikation für einen solchen auftritt. Beim Beispiel des DVD-Players könnte es sich dabei das Umrechnen von komprimierten Video-Daten in unkomprimierte Videodaten handeln. Dabei sind aber nicht die durchgeführten Applikationen als „Benchmark“ zu bezeichnen, wie es häufig der Fall ist, sondern die Performance des Gerätes muss zusätzlich bewertet werden. Es wäre auch sinnlos die alleinige Durchführung einer Applikation, welche im natürlichen Aufgabenbereich des Gerätes liegt, als Bewertung heranzuziehen. [LEV05]

Wie bereits angeführt ist zur Thematik des Energieverbrauchs von elektrischen Geräten nur sehr wenig Literatur zu finden, was wahrscheinlich darauf zurückgeführt werden kann, dass das Thema der Energieverbrauchsoptimierung und des Energiesparens erst in den letzten Jahren verstärkt in den Fokus der Öffentlichkeit gelangt ist. So ist bis vor relativ kurzer Zeit wenig bis gar nicht bekannt gewesen, dass durch das aktive Einsparen von Energie auch sehr viel Geld gespart werden kann.

Einer der wenigen Beiträge auf diesem Gebiet soll allerdings hier nicht unerwähnt bleiben. Es handelt sich um eine Veröffentlichung, in welchem Energieverbrauchsbenchmarks für elektrische Geräte, welche mit Batterie betrieben werden, untersucht werden. Dabei ist einer der zentralen Punkte der Arbeit die Definition einer „Power Benchmark“. Sie wird folgendermaßen definiert: „[A] Power benchmark is defined as a software program that describes the power consumption of the system with respect to a certain stimulus (workload).“ [TUD07] Die Anforderungen an ein solches Programm sehen dabei laut den Autoren der Arbeit so aus: „A power benchmark must be able to distinguish the way power consumption is increasing with workload related to idle state consumption and the type of workload.“ [TUD07]

Die Autoren der Studie haben zur Überprüfung ihrer Methoden mehrere Systeme mit Hilfe des von ihnen entwickelten Benchmark-Programmes untersucht, unter anderem Laptops, PocketPCs und Mobiltelefone. Ihrer Schlussfolgerung nach müssten sich die Systeme nicht nur untereinander an Hand der Benchmarks unterscheiden lassen, sondern man müsste bei mehrmaliger Anwendung auf ein und dasselbe System auch dieselbe Benchmark erhalten. Beide Annahmen wurden in der Arbeit der Autoren bestätigt und belegt. Auch eine Untersuchung ob „ähnliche“ Prozessoren eine ähnliche Charakteristik aufweisen wurde durchgeführt, wobei Prozessoren derselben Bauart aber unterschiedlicher Leistung als „ähnlich“ bezeichnet wurden. Außerdem haben die Autoren eine etwaige Abhängigkeit der Benchmark-Charakteristik von der Umgebungstemperatur untersucht und belegt, dass

eine solche prinzipiell vorhanden ist und dass sich eine höhere Temperatur positiv auf die Batterieleistung auswirken kann. [TUD07]

Als wichtiger Punkt muss herausgestrichen werden, dass es bei den Benchmarks, welche von den Autoren definiert und beschrieben werden, nicht um ganzzahlige Vergleichswerte handelt, sondern um eine Spannungs-Zeit-Funktion beziehungsweise um eine Funktion welche die verbleibende Lebensdauer der Batterie beschreibt handelt. Um zu diesen Funktionen zu gelangen wird das Gerät während drei verschiedener Zeitperioden, auf die später noch genauer eingegangen werden soll, unterschiedlich stark belastet und gleichzeitig die momentane Entladungsrate gemessen. Ein Beispiel wie eine solche Benchmark-Charakteristik in etwa aussehen können, ist in Abbildung 2.2 dargestellt. [TUD07]

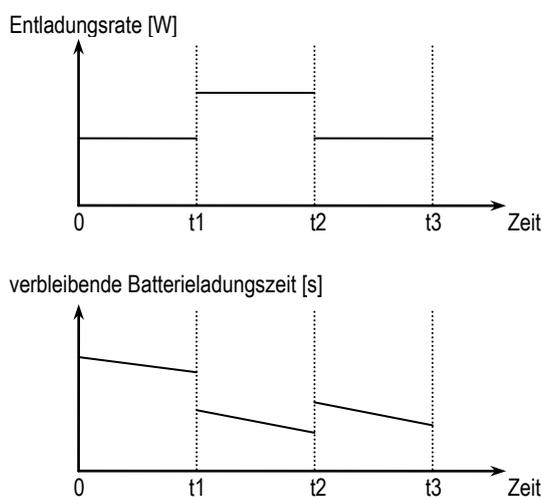


Abbildung 2.2: Benchmark-Charakteristik für batteriebetriebene Geräte [TUD07]

In der ersten Zeitperiode (0 bis t_1) wird die Entladungsrate der Batterie dargestellt, welche im lastfreien Betrieb, also ohne die Ausführung irgendwelcher Programme, auftritt. Danach wird die Entladungsrate während einer Zeitspanne (t_1 bis t_2) dargestellt, in der der Prozessor (oder besser das gesamte Gerät) mit einer möglichst rechenintensiven Aufgabe belastet wird. Zu guter Letzt folgt die Beruhigungsphase (t_2 bis t_3) in der das Gerät wieder unbelastet ist und abgewartet wird, dass die Entladungsrate der Batterie wieder das Niveau erreicht, welches sie am Beginn der Messung hatte. Als charakteristische Zeitverläufe werden sowohl das Spannungs-Zeit-Diagramm angegeben, welches die Entladungsrate der Batterie beschreibt, als auch ein Zeitdiagramm angegeben, welches die übriggebliebene Batterielebensdauer zeigt. Es muss darauf hingewiesen werden, dass die Abbildung nur tendenziell das Aussehen solch einer Benchmark aufzeigt und es sich dabei ausdrücklich nicht um reale Werte sondern um das ideale Aussehen solch einer Charakteristik.

Die Zeitspannen von der ersten Phase genannt Ruhephase, über Belastungsphase und Abklingphase werden entweder fix angenommen („fixed time power benchmark“), wodurch sich der Maximalverbrauch bei einer kurzzeitigen Maximalbelastung darstellen lässt. Oder es soll die so genannte „fixed power consumption values power benchmark“ gezeigt werden, wobei die Zeitspannen variabel sind.

Dabei werden die zu messenden Spannungswerte vorgegeben wodurch eine Abschätzung möglich wird, wie viele Operationen bei einer fixen Energiekonsumrate möglich sind. [TUD07]

Es hat sowohl Vor- als auch Nachteile Funktionsgraphen, wie sie eben beschrieben wurden, als Benchmark zu verwenden. Als Vorteile anzuführen sind eindeutig, dass in einem solchen Graphen zeitliche Zusammenhänge und Tendenzen sehr viel einfacher zu zeigen sind als in ganzzahligen Werten, dadurch ist natürlich der Informationsgehalt der „Benchmark“ um einiges größer. Einschränken muss man aber die eben genannten Punkte dadurch, dass mehr Vorabinformationen beim Benutzer solch einer Benchmark-Charakteristik vorausgesetzt werden muss. Ein weiterer Nachteil bei dieser Methode ist, dass ist die Benchmark nur für batteriebetriebene Geräte definiert und verwendbar ist. Allerdings steigt der Anteil dieser Geräte in nicht zu verachtendem Ausmaß, man denke nur an die Verbreitung von Laptops, Mobiltelefonen und tragbaren Musikplayern, wodurch solch eine Benchmark in Zukunft zu einem wichtigen Instrument zur Bewertung von mobilen, batteriebetriebenen Geräten werden kann.

2.4.3 Fehlende Bewertungskriterien für elektronische Geräte

Auch im Bereich der Benchmarks für elektrische Geräte muss eine zusammenfassende Betrachtung eher ernüchternd ausfallen. Außer den direkt aus den Messungen physikalischer Eigenschaften, wie der Leistungsaufnahme gewonnenen Vergleichswerten, bestehen keine wirklichen Benchmarking-Methoden für diesen Bereich. Dabei wäre es bei der immer stärkeren Verbreitung von Geräten aus dem Computer-, Kommunikations- und Unterhaltungssegment sehr wichtig nicht nur den maximalen Verbrauch von Geräten in eine Bewertung einfließen zu lassen. Während klassische Haushaltsgeräte wie Herd, Backofen und andere auf diese Art und Weise sehr gut in ihrem Verbrauch klassifiziert werden können, reicht das bei Verbrauchern wie Personal Computern, Fernsehern und DVD-Playern nicht aus. Diese Geräte verbrauchen nicht nur während der Zeit in der sie verwendet werden eine nicht unbeträchtliche Menge an Energie. Solange sich das nicht ändert, sollte genau dieser „Stand-By-Verlust“ auch in eine Bewertung miteinbezogen werden, was momentan nicht der Fall ist. Was die Sache noch komplizierter macht ist, dass diese Geräte auch im aktiven betrieb verschieden viel Energie verbrauchen. So verbrauchen beispielsweise Personal Computer unterschiedlich viel Energie bei unterschiedlicher Belastung, was sehr leicht bei der Verwendung von Laptops überprüft werden kann – belastet man das System mit rechenintensiven Aufgaben (z. B. Video-Bearbeitung) sinkt die übrige Batterieleistung dramatisch ab.

Diese Problematik haben auch Tudor und Marcu erkannt und deswegen ein Benchmarking-Programm für mobile, batteriebetriebene Geräte entwickelt. [TUD07] Mit Hilfe dieser Benchmark kann eine Bewertung des Stromverbrauchs für mobile Geräte wie Mobiltelefone, Laptops etc. durchgeführt werden. Leider können damit nicht alle elektrischen Geräte bewertet werden, aber durch die immer stärkere Verbreitung mobile Geräte sind die Einsatzmöglichkeiten schon jetzt beträchtlich.

Auch wenn keine allgemeingültigen Benchmarks für den Energieverbrauch elektrischer und besonders elektronischer Geräte existieren, ist zu erkennen, dass die Problematik der fehlenden Benchmarks im Bereich des Energieverbrauchs langsam erkannt wird.

2.5 Benchmarking auf Chipebene

In diesem Unterkapitel soll ein Blick auf die Chipebene momentaner Computersysteme geworfen werden. Als Beispiel für einen Verbraucher wurden hier deswegen Computersysteme aus allen im letzten Abschnitt besprochenen Verbrauchern gewählt, da hier ein nicht zu verachtendes Einsparungspotential auf der Komponentenebene zu finden ist.

In diesem Bereich finden sich vor allem klassische Benchmarking Verfahren mit Hilfe derer bekannte Systemparameter wie Verarbeitungsgeschwindigkeit und ähnliches bewertet werden können. Allerdings finden sich auch verstärkt Verfahren mit deren Hilfe Eigenschaften wie Energiefluss und die Energieübertragung auf der Ebene der Computerchips verglichen werden können. Mit Computerchips sind hier im Speziellen Mikroprozessoren gemeint, da es sich dabei um eine sehr komplexe Art der integrierten Schaltungen handelt, die auf Grund ihrer Komplexität einen beträchtlichen Energieverbrauch aufweist.

Im ersten Teil des Unterkapitels werden einige bekannte und etablierte Verfahren zum Benchmarken von Computerchips vorgestellt, bevor im zweiten Teil dann einige neuere Methoden diskutiert werden sollen, welche speziell auf die energietechnischen Eigenschaften von Computerchips eingehen.

2.5.1 Benchmarking-Methoden für Mikroprozessoren

Eine der bekanntesten und sicherlich etabliertesten Benchmarks für Mikroprozessoren sind die so genannten IPS (Instructions per Second), oder Befehle pro Sekunde. Bei den heutigen, leistungsfähigen Prozessoren findet meist eine leicht veränderte Vergleichszahl Verwendung, nämlich die Angabe von „Millionen Befehle pro Sekunde“ (Million Instructions per Second). Beide Zahlen geben an wie viele Maschinenbefehle ein Prozessor pro Sekunde abarbeiten kann und sollen ausdrücken wie leistungsfähig ein Prozessor ist. Zur Messung dieser Zahl wird der Prozessor meist mit einem einfach zu implementierenden und gut dokumentierten, aber rechenintensiven Programm belastet. Besonders gut hierfür scheinen sich Probleme aus der Mathematik zu eignen, denn sehr häufig kommen die Aufgaben aus diesem Bereich (Berechnung von Pi, Fibonacci-Sequenz, Primzahlenberechnung etc.). Beispielsweise wird in [MIY88] die Berechnung der Fibonacci-Sequenz verwendet um einen Prozessor zu belasten um dann die Leistungsdaten zu messen.

Einerseits kann dadurch wirklich ein gewisser Eindruck über die Performance gewonnen werden, auf der anderen Seite müsste die Unterschiedlichkeit der Prozessoren eigentlich ebenfalls mit bedacht werden. Man denke hier nur an die beiden total konträren Ansätze Prozessoren zu entwickeln RISC (Reduced Instruction Set Computing) und CISC (Complex Instruction Set Computer). Während die erste Art ein sehr kleines, hoch optimiertes Befehlsset hat, vertraut die zweite auf ein großes Befehlsset mit hochkomplexen Befehlen, welche Aufgaben in nur einem Schritt abarbeiten können für die andere Prozessoren mehrere Berechnungsschritte benötigen. Während also RISC-Prozessoren viele, schnelle Schritte benötigen um eine Aufgabe zu erledigen, werden bei CISC-Prozessoren nur wenige, komplexe und deswegen langsamere Schritte durchgeführt. Ein RISC-Prozessor wird deswegen immer eine höhere MIPS-Zahl haben, ob er deswegen „besser“ als ein CISC-Prozessor einzuschätzen ist, kann aber allein von dieser Grundlage aus nicht gesagt werden. Unter anderem aus die-

sem Grund wurden weitere Benchmarks für die Leistungsfähigkeit von Mikroprozessoren entwickelt.

Der Vergleich der MIPS-Zahl ist wie bereits angedeutet nicht die einzige Art und Weise wie man verschiedene Mikroprozessoren kategorisieren und bewerten kann. So ist es beispielsweise auch möglich die Taktfrequenz als Vergleichskriterium heranzuziehen (natürlich nur wenn es sich um ein synchrones Design handelt). Auf Grund der hohen Werbewirksamkeit dieser Angabe, findet sie sich besonders häufig in eben diesem Bereich Verwendung. Es muss davon abgeraten werden diese Werte als alleiniges Leistungsmerkmal zu verwenden, da sie nur einen relativ kleinen Bereich des Leistungsspektrums eines Prozessors darstellen, ebenso wie die Angabe der Pferdestärken bei einem Kraftfahrzeug.

Obwohl der untersuchte Mikroprozessor nicht mehr dem allerneuesten Stand der Technik entspricht zeigt [MIY88] sehr gut wie die Vorgangsweise bei Performance-Tests von Mikroprozessoren aussehen kann: Man belastet den Prozessor mit einer Reihe von Programmen und misst, beziehungsweise überwacht, die gewünschten Parameter während der Ausführung. In diesem Fall handelte es sich nicht nur um die MIPS, sondern auch um die Gesamtanzahl der Ausführungszyklen, Gesamtanzahl der ausgeführten Befehle, die Registerausnutzung, die Ausnutzung der internen Buffer und etliche andere. Die verwendeten Programme umfassten neben der vorher angesprochenen Berechnung der Fibonacci-Sequenz und der Berechnung von Primzahlen auch ein Programm, welches mit Hilfe des Quicksort-Algorithmus ganzzahlige Werte sortiert. Wie man hier sehr gut sehen kann scheint es bei weitem nicht auszureichen nur die MIPS anzugeben um die Performance eines Mikroprozessors zu klassifizieren. Auch kann man hier sehen, dass nicht jedes Programm den Prozessor auf die gleiche Art belastet, weshalb die Messungen mehrmals mit verschiedenen Belastungsprogrammen durchgeführt werden.

Ein weiterer Vergleich von Prozessoren kann an Hand der so genannten Feature Size erfolgen. Unter der Feature Size versteht man die Größe, die ein Standardelement eines Schaltkreises, zum Beispiel ein Transistor, auf dem Chip einnimmt. Je kleiner diese Zahl ist, desto mehr Transistoren finden auf einem Chip Platz, da dadurch die Packungsdichte vergrößert werden kann. Eine größere Packungsdichte bedeutet in den meisten Fällen auch mehr Funktionalität auf derselben Chipfläche, was allerdings nicht gleichbedeutend mit einer größeren Performance ist. Aus diesem Grund kann auch diese Benchmark alleine nicht ausreichen, da kein direkter Zusammenhang zwischen Funktionalität und Packungsdichte hergestellt werden kann. Im Gegensatz zu Stückpreis und Packungsdichte, wo eindeutig ein Zusammenhang besteht, da auf Grund der teureren Fertigung kleinerer Strukturen auch der Stückpreis höher ist.

Außer dem Ansatz mit welchem die Instruktionen pro Sekunde gemessen werden, haben alle bisher vorgestellte Benchmarks eine sehr starke Fokussierung auf die physikalischen Eigenschaften von Mikroprozessoren. Deswegen soll hier auch auf die so genannten Benchmarking Circuits eingegangen werden. Dass dieser Begriff zwei vollkommen unterschiedliche Verwendungszwecke hat, erleichtert dabei nicht unbedingt das Verständnis.

Einerseits kann als Benchmarking Circuits ein solches Element bezeichnet werden, das eine Benchmark für alle anderen Elemente darstellt, egal ob nun Mikroprozessor oder speziellere Elemente. Ein

Benchmarking Circuit kann als Referenz- und Vergleichsgröße für andere zu überprüfende Prozessoren herangezogen werden.

Andererseits können Benchmarking Circuits dazu dienen, die Wirksamkeit von Design- und Synthesewerkzeugen zu testen. Dazu wird ein Schaltungslayout (in diesem Fall der Benchmarking Circuit) entworfen und an Hand dessen überprüft wie gut dieser Schaltplan von den Synthesetools mit Hilfe von Standardbibliotheken auf einem FPGA, Gate Array oder ähnlichem umgesetzt wird. Benchmarking Circuits in dieser Ausprägung werden beispielsweise bei [LIN99] und [STR00] verwendet.

2.5.2 Energiebenchmarking in der Schaltungstechnik

Um nun auf Energiebenchmarks im Bereich der Computerchips einzugehen muss vorausgeschickt werden, dass im Bereich der Schaltungsentwicklung nicht nur die elektrische, sondern auch die thermische Energie und ihre Auswirkungen auf Bauelemente und Materialien starke Beachtung findet. In den meisten Fällen kann das Energieverhalten im elektrischen Bereich mit dem im thermischen Bereich wenn schon nicht in direktem Zusammenhang so zumindest in einen indirekten Zusammenhang gebracht werden. Das ist deswegen der Fall, weil oft ein starker Anstieg der Temperatur auf einem Mikrochip mit einer starken Anstieg des Stromverbrauchs desselben zusammenhängt. Allerdings wird hier bewusst nicht auf das thermische Verhalten im Sinne von Wärmeleitung und -ableitung eingegangen, da angenommen wird, dass die Abwärme als verlorene beziehungsweise ungenützte Energie möglichst direkt an die Umgebung abgegeben wird.

Ein anderer Bereich, wo sehr viel Energieeinsparung möglich wäre, ist die momentan vorherrschende Schaltungstechnologie bei integrierten Schaltkreisen, den CMOS-Transistoren. Bei dieser gehen große Mengen an Energie als so genannte Leckströme bei den Transistoren verloren. Diese Leckströme treten dabei dann auf, wenn ein als Schalter eingesetzter Transistor die beiden zu schaltenden Kontakte nicht hundertprozentig von einander trennen kann und so immer ein bisschen Strom fließt.

Zwei relativ leicht zu überprüfende, weil physikalische und daher leicht messbare Vergleichszahlen von Prozessoren für die elektrische Energie sind der durchschnittliche Stromverbrauch im unbelasteten Zustand und unter maximaler Last. Letzteres lässt sich dabei sehr leicht mit Hilfe der schon weiter oben behandelten mathematischen Programme simulieren.

Diese beiden eben beschriebenen Vergleichszahlen sind aber im Prinzip auch schon alles was man als Vergleichszahlen für Mikroprozessoren im Energiebereich überhaupt finden kann. So gibt es nirgendwo Richtlinien oder Referenzwerte, welche aussagen dass unter diesen oder jenen Umständen der Energieverbrauch einer digitalen Schaltung nicht einen bestimmten Wert überschreiten darf. Das alleine schon würde eine Möglichkeit darstellen die eigenen Messwerte in Bezug mit einer „offiziellen“ Bezugsgröße setzen zu können. Meist steht man auf dem Standpunkt „je weniger Energieverbrauch umso besser“, was im Fall der unbelasteten Schaltung durchaus Sinn ergibt, bei allen anderen Fällen aber wenig Aussagekraft hat. Nicht umsonst werden bei Performance-Tests die Schaltungen mit verschiedensten Aufgaben belastet.

Abschließend ist zu sagen, dass es im Bereich der Computerchips und Mikroprozessoren ein weites Feld an unterschiedlichen Möglichkeiten gibt die Performance der Objekte zu messen, überprüfen oder zu vergleichen. Was allerdings auffällt ist, dass es relativ wenig „offizielle“ Methodik oder

Vorgaben für solch ein Benchmarking gibt und jedermann sein eigenes Süsschen kochen kann. Besonders auffällig ist dieser Mangel im Bereich der Energie-Benchmarks, wo außer der Messung der rein physikalischen Werte nichts vorhanden zu sein scheint. Eine gute Übersicht und ein Vorschlag wie solch eine Lösung für diese Problematik aussehen könnte (leider wieder weniger auf die Energiedaten- als auf reine Leistungsdaten bezogen) finden sich bei [WEI99] eine andere Übersicht über Benchmarks für die Prozessorleistung findet sich in [WEI90].

2.6 Probleme beim Benchmarking

Das folgende Kapitel soll noch einmal die Probleme, Kritikpunkte und allgemein die Unklarheiten aufzeigen und zusammenfassen, die im Bezug auf Benchmarks im Allgemeinen und im speziellen Fall der Energieversorgung bestehen. Dabei sollen auch die in den vorigen Abschnitten bereits vorgestellten vorhandenen Möglichkeiten und Probleme bei den bestehenden Methoden noch einmal aufgegriffen werden. Außerdem sollen in gewissen Bereichen Vorschläge gemacht werden, inwieweit man das Benchmarking in den einzelnen Bereichen verbessern könnte.

Dabei soll nicht vernachlässigt werden, dass es durch die hohe Diversität, sowohl an Anforderungen als auch an technischen Gegebenheiten, welche in den Bereichen der Energieversorgung und des Energieverbrauchs bestehen, nicht möglich sein kann eine einzige Benchmarking-Technik auf alle Bereiche anzuwenden. Da die Anforderungen in den Bereichen so sehr auseinander gehen, müssen maßgeschneiderte Bewertungstechniken für jeden der Bereiche gefunden werden.

2.6.1 Widersprüche beim Begriff „Benchmark“

Trotz der Unterschiedlichkeit in Dimension als auch Ausdehnung der verschiedenen Bereiche bzw. Schichten kann man bei genauer Betrachtung schon einen einzigen gemeinsamen Nenner finden, der sich durch alle Schichten der Energieversorgung zieht, von der Energiegewinnung bis zur Energiekonsumation in Mikrochips. Bei diesem gemeinsamen Nenner handelt es sich um das verstärkte Interesse an der Reduktion des Energieverbrauchs und aller daraus entstehenden Folgen. Doch selbst solch ein gemeinsamer Dreh- und Angelpunkt muss in den verschiedenen Bereichen unterschiedlich in eine Bewertungsmethode mit einbezogen werden.

Zuerst soll noch einmal genau der Hintergrund des Benchmarkings im Allgemeinen beleuchtet werden, warum es solch eine Verbreitung findet, aber auch welche Missverständnisse es in diesem Bereich gibt.

Bereits der Begriff „Benchmark“ erzeugt einige Widersprüche und einen großen Argumentationspielraum, denn wenn man ihn aus dem Englischen zu übersetzen versucht, kann der Begriff sowohl „Bewertung“ als auch „Bezugsgröße“ bedeuten. Der gar nicht so geringe Unterschied kann folgendermaßen verdeutlicht werden. Eine Benchmark mit der Bedeutung „Bewertung“ wäre beispielsweise eine bei einer Prüfung erhaltene Note, während eine Benchmark im Sinne von „Bezugsgröße“ eine Vorbedingung für eine Lehrveranstaltung wäre, wie z. B. „Um für diese Lehrveranstaltung zugelassen zu werden brauchen die Studenten mindestens ein ‚Befriedigend‘ auf die Einführungslehrveranstaltung.“

Wie man erkennen kann, wäre der erste Schritt bei der Entwicklung einer Benchmarking-Methode eigentlich zu definieren welche der beiden Interpretationsmöglichkeiten mit dem Begriff Benchmark gemeint ist. Meist wird dieser Schritt aber übersprungen, womit der Interpretationsspielraum ziemlich groß wird und Missverständnisse geradezu heraufbeschworen werden, weil es erst einige Zeit dauert bis der Anwender der Methode herausgefunden hat was nun genau die Benchmark eigentlich ist.

Trotz der Probleme der Definition was nun genau mit einer Benchmark gemeint ist, hat sich eine stillschweigende Übereinkunft darüber gebildet was „Benchmarking“ ist. Dabei handelt es sich um eine vergleichende Analyse von verschiedenen Objekten, welche für ein und dieselbe Aufgabe oder Funktion verwendet werden können. Was allerdings bei solch einer Analyse die Benchmark ist, muss aus dem Zusammenhang herausgefunden werden, beispielsweise kann es sich um einen Richtwert handeln an dem die Eigenschaften der Objekte gemessen werden, oder um aus den Eigenschaften der Objekte extrahierte Vergleichszahlen. Beim Benchmarking von Computer-Systemen wird oft noch einen Schritt weitergegangen und das Programm, welches ausgeführt wird um den Prozessor zu belasten, als Benchmark bezeichnet, wobei doch eindeutig das Verhalten des Computer-Systems während der Belastung bewertet werden sollte. Das Programm zur Belastung, ohne eine Überwachung und Bewertung der Reaktion des Systems auf den Input, alleine kann schwerlich eine Bewertung darstellen.

Oft wird deswegen nur von Benchmarks im Allgemeinen gesprochen und das Ganze so dargestellt, als gäbe es die eben beschriebene Begriffsproblematik überhaupt nicht, was entweder auf Unwissen oder auf ein Ignorieren der Problematik zurückgeführt werden muss. Man kann also ohne Übertreibung sagen, dass im deutschen Sprachgebrauch der Begriff Benchmark wohl des Öfteren einfach falsch gebraucht wird, denn egal welche der beiden Bedeutungen man meint, die alleinige Angabe einer Eigenschaft als Benchmark, ist einfach falsch. Eine Benchmark muss sich immer auf einen anderen Wert beziehen oder zumindest die Bezugsgröße für andere Werte darstellen.

Zusammenfassend kann also gesagt werden, dass es sich bei einer Benchmark um eine Vergleichszahl oder um ein Set charakteristischer Werte handelt, welche durch eine Verallgemeinerung und Bewertung einzelner Eigenschaften bzw. durch die zusammenfassende Bewertung mehrerer Eigenschaften eines Objektes erhalten wurde. Diese Vergleichszahl kann nun dafür verwendet werden, um mehrere Objekte untereinander zu vergleichen, wobei es sich um Objekte handeln muss, die grundsätzlich an Hand der ausgewählten Eigenschaften vergleichbar sind. Oder man kann die Vergleichszahl heranziehen und mit einer bestehenden Vergleichszahl oder einem bestehenden charakteristischen Werteset verglichen werden kann. Eine Benchmark alleine hat also grundsätzlich wenig bis gar keine Aussagekraft, außer der Anwender hat Vorwissen darüber welcher Art Qualität die Benchmark haben muss, um aus der erhaltenen Benchmark herauslesen zu können ob man die Eigenschaften des Objekts als „gut“ oder „schlecht“ einstufen kann.

Bei all diesen Überlegungen wurde noch kein Wort zur Aussagekraft verschiedener Benchmarks gesagt. In den meisten Fällen ist es um diese Aussagekraft eher schlecht gestellt. Man denke nur an die Angabe der Motorleistung (gemessen in Pferdestärken) von Kraftfahrzeugen. Würde man nur diese Angabe als alleiniges Kriterium zur Bewertung der Kraftfahrzeuge untereinander heranziehen und festlegen, dass Kraftfahrzeuge mit einer hohen Motorleistung als „gut“ zu bewerten sind, so

müsste man zu dem Schluss kommen, dass Autobusse und Lastkraftwagen eindeutig und immer als gut zu bewerten sind. Das mag für eine eingeschränkte Aufgabenstellung durchaus zutreffen, doch sicherlich nicht für jede Person, die ein Kraftfahrzeug kaufen will. Wie man sieht muss auch hier eine Einschränkung getroffen werden, wobei es möglich ist diese Einschränkung vorab zu treffen („Es sollen nur Sportwagen verglichen werden“) oder durch die Ergebnismenge durch zusätzliche Bewertungskriterien in die gewünschte Richtung gelenkt wird, in unserem Beispiel durch zusätzliche Bewertung der Höchstgeschwindigkeit.

Das Beispiel mit den Kraftfahrzeugen mag etwas an den Haaren herbeigezogen wirken, doch gibt es etliche Bereiche in denen solche sehr eingeschränkten Kriterien wie die Motorleistung als schlagendes Verkaufs- und Kaufargument herangezogen werden. So ist eines der Hauptargumente beim Kauf von Personal Computern die maximale Taktrate des Hauptprozessors. Meist wird suggeriert, dass für jegliche Einsatzgebiete jene Computer am besten sind, die die höchste Taktrate aufweisen, ungeachtet des zukünftigen Einsatzgebietes. Wozu braucht aber z. B. ein Konsument, dessen Interesse darin besteht einen Internet-fähigen Computer zu besitzen um hin und wieder im weltweiten Netz zu surfen, das neueste High-End-Modell? Eine Verbesserung dieser irreführenden Situation ist allerdings nicht in Sicht. So zeichnet sich zwar in den letzten Jahren ein Abkehren von der Entwicklung immer höherer Prozessortakraten ab, doch in der Werbung geschieht eine reine Verschiebung, waren früher die Taktraten der Mainboards das schlagende Verkaufsargument, so sind es heute die leistungsfähigsten Grafikkarten.

2.6.2 Überblick über die Bewertungsmethoden für den Energieverbrauch

Nach diesen Überlegungen zum Begriff Benchmark soll noch einmal auf die Problematik des Energieverbrauchs-Benchmarking eingegangen werden. In dieser Arbeit wurde die gesamte Energieversorgung in insgesamt vier Schichten geteilt, vom Kraftwerk bis zu den Computerchips, die in letzter Instanz Strom verbrauchen. Neben der bereits erwähnten Schicht der Computerchips handelte es sich bei diesen Schichten um den Bereich der elektrischen Geräte, der Endkunden und eben den Bereich der Energieversorgung. Diese Zusammenfassung hier soll noch einmal die wichtigsten Kritikpunkte, die in jenen Bereichen aufgetreten sind aufgreifen und in einer kompakten Form zusammenfassen.

Beginnend bei dem Bereich den die großen Energieversorgungsunternehmen dominieren, fällt auf, dass gerade hier wenige Benchmarks existieren, die das Hauptaugenmerk auf Energieeinsparung legen. Das kann nun insofern begründet werden, dass sich die Energieversorger eher um die Sicherheit ihres Netzes, als um die Einsparungen kümmern, was sicherlich für die Betreiber der Stromnetze die primäre Sorge ist, ist doch der Anspruch ihrer Kunden zu jeder Zeit genügend elektrische Energie zu haben. Aus diesem Grund wurden auch Benchmarks entwickelt durch welche die Versorgungssicherheit bewertet werden kann. Man könnte zwar annehmen, dass es in den umfangreichen Regelmethodiken der Energieversorger gewisse Bewertungs- und Prognosemethoden gibt, allerdings beschäftigen sich diese eher mit dem großen Ganzen („Wie viel Energie wird in Wien um 12:00 gebraucht?“) und nicht mit den Effekten und Folgen des Energiesparens individueller Haushalte oder Personen.

Gerade hier sollte aber ein Umdenkprozess eintreten, denn durch die Liberalisierung und Bestrebung der Konsumenten immer größere Einsparungen zu erreichen werden sich die Anforderungen an die

Energieversorger sehr verändern. Ein Schritt in diese Richtung wurde auch gezeigt, so wurden bereits Modelle für das Verhalten des gesamteuropäischen Energienetzes entwickelt um die hochdynamischen Vorgänge, die durch den Handel mit Energie entstehen, modellieren zu können. Die dabei verwendeten Benchmarks sind allerdings ein bisschen missverständlich definiert, denn es wird dabei das Verhalten eines Stromnetzes vereinfacht dargestellt, und nicht bewertet. Bewertet werden sollen mit diesen Modellen verschiedene Methoden Lastflüsse zu steuern.

Außerdem existieren im obersten Segment der Energieversorgung Benchmarks, welche helfen die Effizienz beziehungsweise Funktionsweise von Stromerzeugern zu bewerten. In diesem Bereich wurde eine Benchmark zur Bewertung von verschiedenen Topologien von Windenergieanlagen vorgestellt.

Anders ist die Situation bei den Endkunden. Hier findet immer mehr ein Umdenken und eine Änderung in Richtung energieeffizientes und ökologisch korrektes Verhalten statt. Umso erstaunlicher ist, dass es sehr wenige wirklich leistungsfähige Benchmarking und Bewertungsmethoden für diese Sparte gibt. Positiv herausgestochen hat dabei ein Angebot eines österreichischen Energieversorgers, der über ein Internet-Angebot eine individuelle Bewertung des Stromverbrauchs von Privatkunden anbietet. Was dabei auffällt ist, dass der Benutzer, bevor er eine Bewertung und Optimierungsvorschläge erhält, sehr detailliert über die im Haushalt vorhandenen Energieverbraucher, den Zustand des Hauses und das Energieverbrauchsverhalten befragt wird. Erst danach wird eine Bewertung und Optimierung durchgeführt. Hier wird offensichtlich was sonst bei vielen der Benchmarking-Methoden nicht beachtet wird, nämlich der Umstand, dass sehr viele Umgebungsvariable notwendig sind, um eine wirklich aussagekräftige Benchmark zu erhalten.

Was uns schon zu den in Kapitel 2.4 vorgestellten Methoden des Benchmarkings bringt. Dabei handelt es sich um Methoden, welche die Performance von elektrischen Geräten, in unserem Fall von Computersystemen bewerten. Computersysteme wurden deswegen gewählt, weil sie im Unterschied zu anderen Verbrauchern eine hochdynamische Energieverbrauchscharakteristik aufweisen, die Verbreitung von solchen System in den letzten Jahren sehr zugenommen hat und dieser Trend sich in Zukunft noch verstärken wird. Hinzu kommt, dass die Verbraucher aus dem Bereich der Unterhaltungs- und Kommunikationselektronik bereits jetzt jene sind, auf deren Konto ein Großteil der verbrauchten Energie geht.

Hier wurde deutlich, dass auch es in diesem Bereich an Bewertungsmethoden fehlt, welche den Energieverbrauch von elektrischen Geräten klassifizieren. Es existieren eine Menge von Methoden um die Leistungsfähigkeit von elektronischen Geräten zeigen zu können, aber keine Methode klärt auf hinreichende Art und Weise ob ein Gerät energiesparend arbeitet oder nicht. Allerdings konnte auch hier eine relativ neue Veröffentlichung gefunden werden, in der rumänische Wissenschaftler eine Benchmark-Charakteristik für den Energieverbrauch von batteriebetriebenen, mobilen Geräten vorstellen. Diese Charakteristiken sind zwar nicht als klassische Benchmarks im Sinne von Vergleichszahlen aufzufassen und zu verwenden, allerdings können sie eindeutig als Schritt in die richtige Richtung gesehen werden.

Ähnlich wie in der „übergeordneten“ Schicht der Energieverbraucher ist die Situation bei Computerchips und anderen integrierten Schaltungen. Diese sind deswegen hier behandelt worden, weil sie beinahe schon in jedem Verbraucher des Haushalts in irgendeiner Weise präsent sind, sei es nun in

der Steuerung des Mikrowellengerätes oder der Hauptprozessor des Home-Entertainment-Systems. Obwohl diese Komponenten beinahe überall zu finden sind, ist der Energieverbrauch dieser Schaltungen noch immer eine der Eigenschaften, welche erst in zweiter Linie optimiert wird. In erster Linie werden meist die offensichtlichen Merkmale wie Geschwindigkeit oder Integrationsdichte erhöht. Dabei wäre es an der Zeit auch hier den Hebel anzusetzen, geht doch ein großer Teil des verbrauchten Stroms als „ungenutzter“ Leckstrom in den Transistoren der Schaltungen verloren. Und doch begnügt man sich auch hier mit der Bewertung allgemeiner Leistungsmerkmale und der recht ungenügenden Messung des Gesamtstromverbrauchs im unbelasteten und im belasteten Zustand. Auch hier fehlen ganz eindeutig Bewertungskriterien und -methoden.

2.6.3 Notwendige Entwicklungen im Bereich der Energiebenchmarks

Ein großes Thema dieser Arbeit sind die Schwächen, welche bei diversesten Benchmarking-Methoden bestehen und eine Benchmark mag auch nicht in allen Fällen das idealste Mittel sein um die Bewertung verschiedenster Zusammenhänge durchführen zu können. Trotzdem können gute Benchmarks ein nicht zu unterschätzendes Hilfsmittel sein. Im Gegensatz darf natürlich nicht vergessen werden, dass mit einer Benchmark auch sehr viel Unfug betrieben werden kann, schließlich ist es oft so, dass die alleinige Angabe einer Benchmark oft als einziges Argument für oder gegen das zu bewertende Objekt herangezogen wird.

Deswegen ist es wichtig bei der Verwendung einer Methode zum Benchmarken von Objekten immer die genauen Definitionen und Eigenschaften dieser Benchmark zu kennen. Genauso ist es sicherlich von Vorteil, wenn vor dem Anwenden der Benchmarking-Methode hinterfragt wird, was damit eigentlich erreicht werden soll und welche Ergebnisse zu erwarten sind, denn das falsch interpretierte Ergebnis einer Benchmarking-Methode kann mehr Schaden anrichten als wenn man auf den Einsatz der Methode verzichtet hätte.

Es kann keine ideale Benchmark geben, die Methoden sollen und müssen immer auf das genaue Einsatzgebiet angepasst und optimiert werden und auch nach diesem Schritt sind die Ergebnisse mit höchster Vorsicht zu genießen. Denn eine Benchmark ist immer eine absichtlich vorgenommene Verallgemeinerung der komplizierteren wahren Begebenheiten. Dadurch können und werden natürlich einige Eigenschaften der Objekte vernachlässigt, die sich in anderen Situationen vielleicht als essentiell herausstellen können.

Daher wäre es wichtig, dass immer auch die Begleitumstände und genauen Kriterien der Benchmarks angegeben werden, doch genau darauf wird oft, aus verschiedensten Gründen verzichtet. Einerseits werden gerade im Bereich wo es um Profite geht aus Gründen der Werbewirksamkeit immer nur die positiven Benchmarks angegeben und alles weitere verschwiegen, andererseits haben die Verantwortlichen die Sorge mit zu vielen Details die Zielgruppe zu verwirren. Doch genau hier liegt ein Denkfehler, denn grundsätzlich stimmt es natürlich, dass zu viele Details leicht überfordernd wirken, doch sollte an Hand der angegebenen Benchmarks ein Interesse geweckt werden bei welchen der Objekte eine genaue Untersuchung der Eigenschaften zielführend sein kann.

Darüberhinaus fehlt es in allen Bereichen der Energieversorgung an effizienten und aussagekräftigen Benchmarking-Methoden, die die Veränderungen des Energiemarkts auf effiziente und optimale Weise unterstützen können. Hier ist die Wissenschaft gefragt diese Methoden zu entwickeln, damit

es möglich ist das Energieverbrauchsverhalten aller Personen und Objekte zu optimieren, welche auf dem weitläufigen Energiebereich eine Rolle spielen. Auch wenn die Entwicklung und Etablierung dieser Methoden sicherlich von Kosten begleitet sind und es einige Zeit dauern kann bis sie sich durchgesetzt haben, ist der Nutzen der daraus lukriert werden kann sicherlich um einiges größer. Auch die Politik ist hier gefragt die geeigneten Voraussetzungen zu schaffen, dass die Einsparung von Energie auf kurze wie auf lange Sicht belohnt wird.

3. Modellierung von Energieverbrauchern

Um eine umfassende Analyse und Verarbeitung von Energieverbrauchsdaten durchführen zu können, ist es unerlässlich zu wissen welche Stromverbraucher im zu untersuchenden Objekt, sei es nun ein Geschäft oder eine andere Ansammlung von Verbrauchern, verwendet werden. Da es zu aufwändig und teuer wäre den Stromverbrauch jedes einzelnen dieser Verbraucher durch Strommessgeräte und andere Messtechnik zu erfassen, muss für jeden Verbrauchertyp ein Modell vorhanden sein, mit Hilfe dessen der Stromverbrauch für einen Tag modelliert werden kann um eine ungefähre Vorstellung davon zu erhalten wie sich der Verbraucher an einem bestimmten Tag verhalten haben könnte.

Das folgende Kapitel widmet sich der Entwicklung und Erstellung von Modellen um den Stromverbrauch einzelner Energieverbraucher während eines Tages zu modellieren und dadurch den Verbrauch eines gesamten Geschäftes abschätzen zu können. Dazu werden zuerst die vorhandenen Daten, Modelle und bestehende Vorgangsweisen zur Bewertung des Stromverbrauchs einzelner Geschäfte analysiert.

Desweiteren werden bereits vorhandene Modelle mit realen Daten verglichen. Dabei wird gezeigt, dass einige dieser Modelle zu stark von der Realität abweichen und somit eine Neuentwicklung dieser Modelle notwendig ist. Zuletzt wird untersucht inwieweit eine Verbesserung der Modelle erreicht werden konnte.

3.1 Bestehende Vorgangsweisen bei der Datenanalyse

Bei den vorhandenen Daten einer Kaufhauskette handelt es sich wie bereits erwähnt um die Lastprofile und die Messdaten einzelnen Energieverbraucher. Für jeden dieser Energieverbraucher liegen umfangreiche Messdaten vor aus denen Modelle für die Verbraucher erstellt werden sollen. Da die Messgeräte eine Sample-Rate von 4 Messzeitpunkten pro Stunde haben setzt sich ein Tagesverlauf aus 96 Messwerten zusammen.

Um den Stromverbrauch einer Filiale zu analysieren wurde bisher folgendermaßen vorgegangen: Es wurden die Betriebsdaten der Stromverbraucher verwendet um Modelle ihres Tagesverlaufs zu generieren (im Abschnitt 3.2 werden diese Modelle einer genaueren Prüfung unterzogen). Aus diesen Modellen wurde ein Gesamtmodell für den Stromverbrauch einer Filiale durch Addition aller vorhandenen Modelle erzeugt.

Da sich der Stromverbrauch eines Geschäftes an Sonn- und Feiertagen sehr von dem an anderen Tagen unterscheidet und dadurch an (zumindest) einem Siebtel aller Tage ein nicht unbeträchtlicher Fehler in Kauf zu nehmen wäre, musste auch diesem Umstand Rechnung getragen werden. Um auch für Sonn- und Feiertage zu korrekten Ergebnissen zu kommen wurde davon ausgegangen, dass einige der Verbraucher (wie z. B. die Beleuchtung) an diesen Tagen nicht eingeschaltet wurden. Es wurden also für jeden Verbraucher zwei Modelle erstellt. Aus dem in Abbildung 3.1 gezeigten Wochenstromverlauf einer Filiale ist sehr deutlich der Unterschied zwischen Arbeitstagen, mit den 6 ausgeprägten Spitzen zwischen 340 und 470 kWh, und dem Sonntag, mit einem Stromverbrauch um die 150 kWh, zu erkennen.

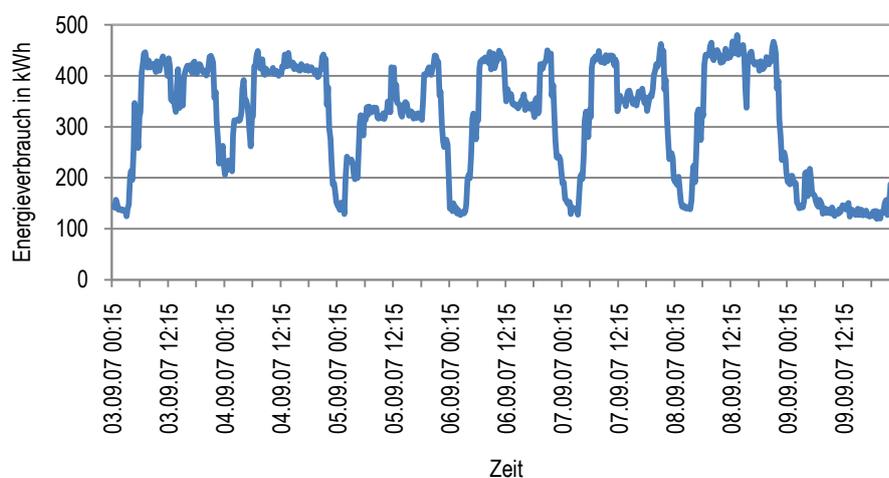


Abbildung 3.1: Wochenstromverlauf eines Geschäftes im September 2007

Aus den nun vorhandenen zwei Modellen, jenem für normale Arbeitstage und jenem für Sonn- und Feiertage, konnte ein Modell des Stromverbrauchs einer ganzen Woche erstellt werden. Bei der Auswahl der Daten einer Woche, welche zur Grundlage der Analyse werden soll, wurde darauf geachtet, dass es sich um eine „normale“ Woche, also mit 6 Arbeitstagen, gefolgt von einem Sonntag handelt. Das Wochenmodell wurde dann erstellt aus 6-mal dem Modell für einen Arbeitstag und einmal dem Modell für einen Sonntag.

Hier ist eine kleine Anmerkung notwendig: Da es sich hier um Daten einer Filialkette aus Deutschland handelt kann der Samstag wie ein normaler Geschäftstag behandelt werden, da in Deutschland die Öffnungszeiten auch am Samstag nicht eingeschränkt sind.

Um nun reale Daten einer Filialen eines Monats zu überprüfen wurden die Werte einer Woche, welche in diesem Beobachtungszeitraum liegt, genommen und mit dem Modell verglichen. Dazu wurde die Differenz zwischen Realverlauf und Modell gebildet. Aus diesen Differenzwerten wurde danach eine Summe gebildet, die als Abweichung vom Sollverlauf aufgefasst wurde. Bei einem positiven Wert war der Stromverlauf höher als erwünscht, bei einem negativen Wert wurde gespart. Um die Analyse, welche händisch durchgeführt werden musste, zu vereinfachen wurde dieser Wert nur für eine Woche berechnet und, um auf einen Wert für den gesamten zu analysierenden Zeitraum zu

kommen, mit 4 (wenn es sich um ein Monat gehandelt hat) multipliziert. Es wurde also aus dem Stromverbrauchsverhalten einer Woche auf einen größeren Zeitraum geschlossen, ungeachtet ob solch ein Schluss wirklich zulässig war oder nicht. Hier wurde also weder darauf geachtet ob der Stromverlauf in den folgenden Wochen ungefähr dem der untersuchten Woche entsprochen hat, noch ob vielleicht irgendwelche Feiertage das Ergebnis zusätzlich verfälschen hätten können.

Aus diesem Ergebnis und einer manuellen Plausibilitätsprüfung des Stromverlaufes in der gewählten Woche („Können diese Messdaten denn überhaupt stimmen?“) wurde dann eine detaillierte Analyse erstellt, welche das Einsparungspotential für diese Filiale und Ratschläge zur Stromkostenreduktion enthält.

Die größten Schwächen der oben vorgestellten Vorgangsweise sind dabei folgende:

- Es wird vom Stromverbrauch einer (beliebigen) Woche auf einen größeren Beobachtungszeitraum geschlossen, was selbst wenn es sich nur um einen einzelnen Monat gehandelt hat, zu großen Abweichungen führen kann.
- Solch eine Analyse per Hand durchzuführen ist in höchstem Maße fehleranfällig und bei der Analyse mehrerer Filialen sind die Arbeitsschritte genau die gleichen, was für den Ersteller solch einer Analyse sehr monoton sein kann.
- Es erfordert sehr viel Fachwissen um aus dem Stromverlauf einer Woche „sehen“ zu können ob der Stromverbrauch den Vorgaben entsprochen hat, oder Abweichungen vorliegen, die nicht der Regel entsprechen.

3.2 Vergleich vorhandener Modelle mit realen Daten

Wie im vorherigen Unterkapitel erwähnt wurde, wurden bei der bisherigen Vorgangsweise Modelle für die einzelnen Verbraucher verwendet. Da aber die bisherigen Analysen weitgehend „per Hand“ erstellt wurden, lag das Hauptaugenmerk bei der Erstellung der Modelle nicht in Detailgenauigkeit, sondern darin, dass diese Modelle leicht zu händeln waren.

Bei den meisten der früher verwendeten Modellen wurde davon ausgegangen, dass die Verbraucher zu Beginn der Öffnungszeiten eingeschaltet werden und am Ende der Öffnungszeiten wieder ausgeschaltet wurden. Zwischen diesen beiden Zeitpunkten wurde angenommen, dass der Stromverbrauch konstant ist. Diese Modelle sollen im Folgenden vorgestellt und untersucht werden.

3.2.1 Modell für eine Kälteanlage

Als Kälteanlagen werden alle in einem Geschäft vorhandenen Geräte zur Kühlung von Lebensmitteln bezeichnet und zusammengefasst. Dabei kann es sich um separate Kühltruhen und –schränke, welche alle eigenen Kühlelemente aufweisen, oder um in Kühlkreisen organisierte Kühlregale mit wenigen zentralen Kühlpumpen handeln.

Um die Analyse der Energiedaten möglichst schnell und unkompliziert zu gestalten wurde der Stromverbrauch von Kälteanlagen auf sehr einfache Weise modelliert. Dabei nahm man bewusst stark von der Realität abweichende Werte in Kauf, obwohl bekannt war, dass gerade die von der Temperatur stark beeinflussten Kälteanlagen sehr starke Schwankungen aufweisen können. Im früheren Modell wurde davon ausgegangen, dass der Verbrauch der Kälteanlagen in der Nacht durch isolierende Rollos auf einem konstanten Wert gehalten wird und diese Rollos zu Beginn durch die Mitarbeiter geöffnet und am Ende der Öffnungszeiten geschlossen werden. Zwischen diesen Zeitpunkten wurde ein konstanter Verbrauch, der höher als jener in der Nacht ist, angenommen. Eine Darstellung dieses Modells findet sich in Abbildung 3.2.

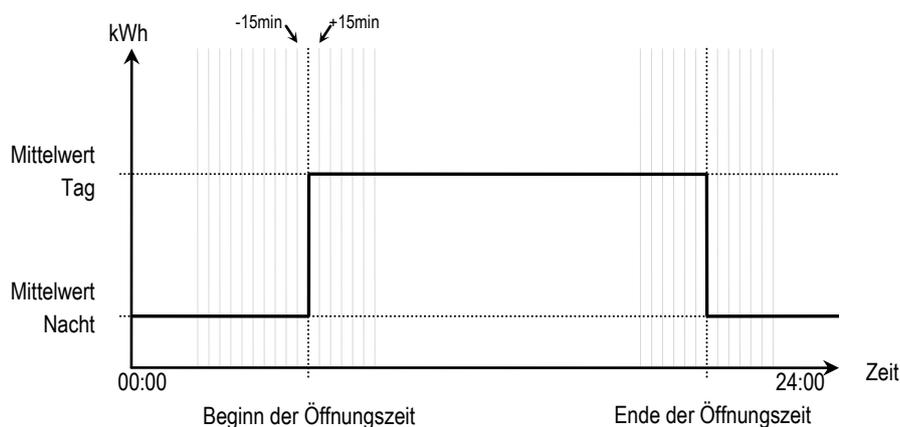


Abbildung 3.2: Modell einer Kälteanlage

3.2.2 Modell für eine Lüftungsanlage

Im Gegensatz zur weit verbreiteten Annahme, dass eine Lüftungsanlage dieselben Funktionen erfüllt wie eine Klimaanlage besteht ein eindeutiger Unterschied zwischen den beiden Gerätetypen. Während eine Klimaanlage die Lufttemperatur aktiv senkt, handelt es sich bei einer Lüftungsanlage um ein Gerät, welches die vorhandene Luft in einem Raum mit frischer Luft von außerhalb des Gebäudes austauscht. Da die automatische Aktivierung der Lüftung von der Differenz zwischen Außen- und Innentemperatur abhängig gemacht wird, handelt es sich um einen stark thermisch beeinflussten Prozess.

Um den Stromverbrauch einer Lüftungsanlage gut modellieren zu können muss erklärt werden, dass Lüftungsanlagen mehrere verschieden leistungsstarke Stufen haben, deren Effizienz allerdings nicht steigt, sondern sinkt. So tauscht Lüftungsstufe 2 im Gegensatz zu Lüftungsstufe 1 zwar doppelt so schnell die Luft aus, verbraucht aber 4-mal so viel Strom. Ließe man nun die Lüftung auf Stufe 1 doppelt so lange eingeschaltet wie auf Stufe 2 hätte man den selben Effekt (die selbe Menge Luft wurde getauscht), aber man hätte 50% der Energie eingespart. Aus diesem Grund wird als erster Schritt zur Senkung des Energieverbrauchs und für einen effektiven Einsatz der Lüftung geraten auf den Einsatz anderer Stufen als der ersten zu verzichten.

Wie im Fall der Kälteanlage wurde auch hier ein stark vereinfachtes Modell verwendet, welches in Abbildung 3.3 abgebildet ist. Für dieses Modell wurde angenommen, dass die Lüftung weitgehend optimal läuft, wenn sie 1-1,5 Stunden Beginn der Ladenöffnungszeiten Belüftungsstufe 1 aktiviert wird, während der Geschäftszeiten die Lüftung permanent auf dieser Stufe aktiviert bleibt und am Ende der Geschäftszeiten deaktiviert wird. Für den Energieverbrauch der ersten Lüftungsstufe wurde ein konstanter Wert verwendet.

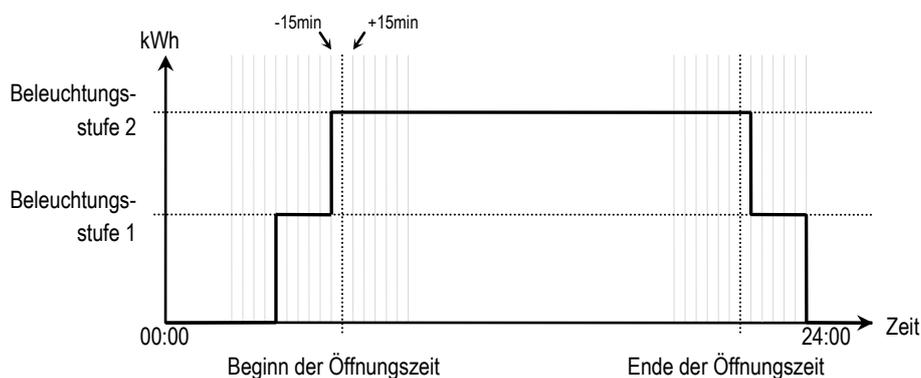


Abbildung 3.3: Modell einer Lüftungsanlage

Die leistungsstarken höheren Belüftungsstufen gehen auf Grund ihres hohen Energieverbrauchs nicht in das Modell ein, da diese wie bereits erwähnt sehr ineffizient sind und daher meist empfohlen wird diese überhaupt nicht, oder nur im äußersten Notfall zu verwenden.

3.2.3 Modell für die Beleuchtung

Das bisherige Modell der Beleuchtung eines Geschäftes umfasst die gesamte vorhandene Beleuchtung. Es wird dabei angenommen, dass in jedem Objekt, das analysiert werden soll zwei verschiedenen Beleuchtungsstufen vorhanden sind. In Abbildung 3.4 ist das bisher verwendete Modell des Stromverbrauchs dargestellt.

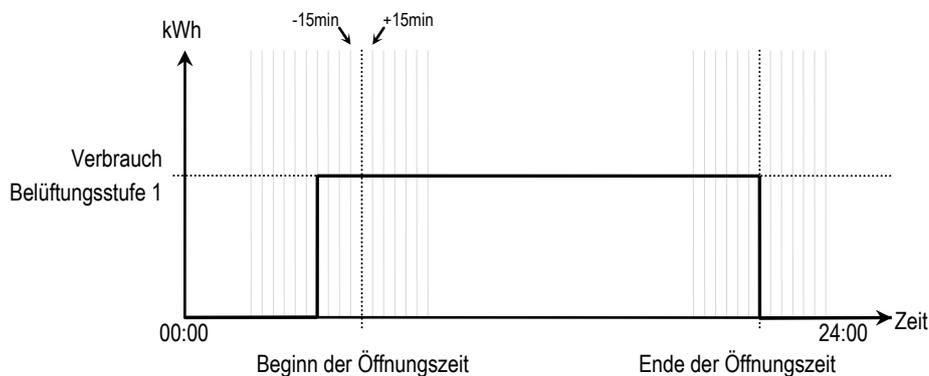


Abbildung 3.4: Modell für die Beleuchtung

In der dieser Abbildung sind sehr gut die beiden zu Einsatz kommenden Beleuchtungsstufen und der daraus entstehende Stromverlauf zu erkennen. Wie man sehr gut sehen kann wird die erste der beiden Stufen meist automatisch kurz bevor die ersten Mitarbeiter in das Geschäft kommen eingeschaltet wird. Die zweite Beleuchtungsstufe wird nur während der Öffnungszeiten verwendet und wird deswegen ungefähr 15 Minuten vor Beginn ein- und 15 Minuten nach Ende der Öffnungszeiten ausgeschaltet. Während der Öffnungszeiten werden also beide Beleuchtungsstufen verwendet, um eine optimale Beleuchtung des Geschäftes zu gewährleisten.

Hier ist zu erwähnen, dass keine der beiden Beleuchtungsstufen als ganzes eingespart werden kann, das das Beleuchtungskonzept eines Geschäftes darauf ausgelegt ist, dass eben nur die beiden Beleuchtungsstufen zusammen eine Filiale ausleuchten können und keine der beiden Beleuchtungsstufen alleine ausreicht.

Nach Ende der Öffnungszeiten brennt das Licht auf Stufe 1 noch einige Zeit weiter, was meist auf Grund von Werbezwecken geschieht, oder deswegen, weil in dieser Zeit die Regals neu bestückt werden, oder die Reinigung des Geschäftes vorgenommen wird. Danach wird das Licht (meist automatisch mittels einer Zeitschaltuhr) abgeschaltet.

Während der Nacht, an Sonntagen und immer wenn sich niemand im Geschäft aufhält und deswegen keinerlei Beleuchtung notwendig ist soll das Licht weder auf Stufe 1 noch auf Stufe 2 aufgeschaltet sein.

3.2.4 Modell für die Grundlast

Alle anderen Verbraucher von denen bekannt ist, dass sie vorhanden sind und welche keine spezielle Signifikanz in der Höhe des Verbrauchs oder des zeitlichen Verbrauchsprofils aufweisen, werden in der so genannten Grundlast zusammengefasst. Dabei wird angenommen, dass es einen konstanten Verbrauch außerhalb der Öffnungszeiten und ebenso einen konstanten, aber höheren, Verbrauch während der Öffnungszeiten gibt. Das Modell für die Grundlast ist Abbildung 3.5 dargestellt. Durch die Verwendung solch eines Modells ist es möglich auch solche Verbraucher mit in die Beobachtung zu integrieren, die nur in einigen wenigen oder überhaupt nur in einem Geschäft vorhanden sind. Dadurch konnten auch gänzlich untypische Verbraucher in die Vergleiche einbezogen werden.

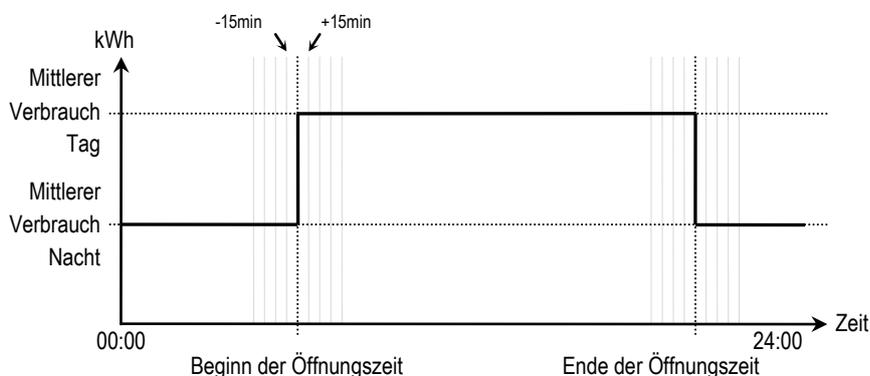


Abbildung 3.5: Modell für die Grundlast

3.2.5 Unterschiede zwischen Modell und Realität

Um nun einschätzen zu können wie gut oder schlecht die bereits vorgestellten Modelle reale Daten abbilden können, werden hier die Modelle für Kälteanlagen und Lüftungsanlagen realen Messwerten gegenübergestellt. Für die Beleuchtung liegen keine realen Messwerte vor. Stattdessen liegen im Fall der Beleuchtung die Schalterzustände vor. Bei einer Überprüfung dieser Werte für die Beleuchtung wurde festgestellt, dass sie weitgehend dem in Abbildung 3.4 dargestellten Modell entsprechen. Auch beim Modell der Grundlast gibt es keinen Bedarf einen Vergleich anzustellen, da es sich dabei um eine Menge von immer verschiedenen Verbrauchern handelt und so die Zusammensetzung von Filiale zu Filiale zu verschieden ist, als dass solch ein Vergleich zielführend sein kann.

Kälteanlage

Bei dem als Kälteanlage bezeichneten Verbraucher handelt es sich um alle in einem Geschäft vorhandenen Kühlregale und -truhen, meist sind diese durch einen Kühlkreis mit einer oder mehreren Kühlturmpumpen gekoppelt.

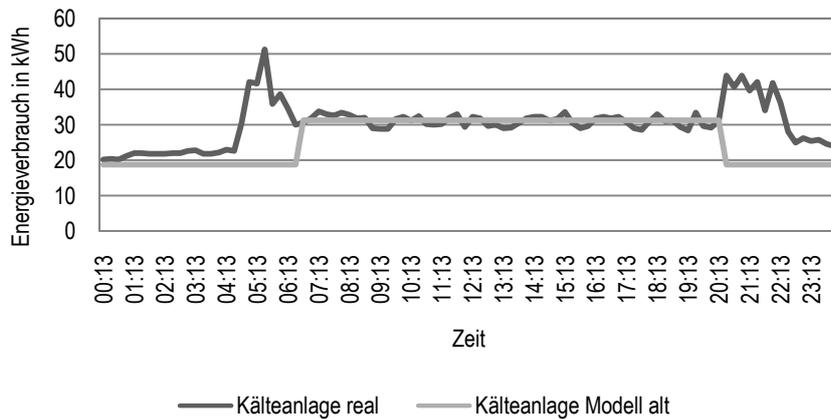


Abbildung 3.6: Modell und Realdaten einer Kälteanlage, Filiale 1

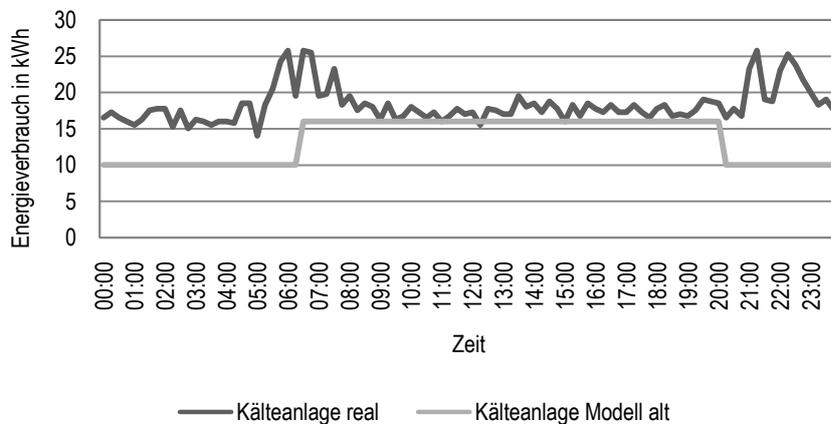


Abbildung 3.7: Modell und Realdaten einer Kälteanlage, Filiale 2

Bei einer Gegenüberstellung der Daten der Kühlanlagen zweier Filialen mit den jeweils zugehörigen Modellen kann man sofort erkennen, dass eine sehr große Fläche vor dem Anfang und nach dem Ende der Öffnungszeiten keine Repräsentation im Modell hat (siehe Abbildung 3.6 und Abbildung 3.7). Auch die ungewöhnlichen hohen Werte, welche viel höher als der Tagesdurchschnitt sind und ebenfalls vor und nach den Geschäftszeiten auftreten, werden im Modell nicht dargestellt. Das Modell der Kälteanlage bedarf also eindeutig einer weitgehenden Neuentwicklung.

Lüftungsanlage

Um das Problem beim Modell der Lüftungsanlage aufzuzeigen, ist es nicht ausreichend zwei Stichproben beliebiger Tage zu nehmen. Stattdessen ist es notwendig die Messdaten von zwei Tagen in verschiedenen Jahreszeiten (Sommer und Winter) zu untersuchen. Das ist deswegen der Fall, da die Lüftungsanlage und ihre Funktion sehr stark von der Außentemperatur und deswegen vom Jahreszeitenwechsel abhängig ist. Deswegen wurden in den Abbildung 3.8 und Abbildung 3.9 jeweils die Realdaten zweier Tage dem Modell gegenübergestellt.

Dabei ist auffällig, dass besonders im Winter der Stromverbrauch drastisch erhöht ist, was darauf schließen lässt, dass statt der Lüftungsstufe 1 die Lüftungsstufe 2 verwendet wurde. Wie bereits erklärt tauscht eine Lüftungsanlage bei Betriebsstufe 2 doppelt so schnell die gleiche Luftmenge wie bei Stufe 1 aus. Allerdings ist der Energieverbrauch dann nicht doppelt so hoch wie bei Stufe 1, sondern 4 Mal so hoch, da der Zusammenhang zwischen der Temperatur und dem Stromverlauf in diesem Fall potentiell und nicht multiplikativ ist. Auf Grund dessen wird meist empfohlen nur Stufe 1 zu verwenden und bei Bedarf diese länger laufen zu lassen. Den Umstand, dass sich der Stromverbrauch im Gegensatz zum Modell in diesem Fall beinahe vervierfacht kann man sehr gut in den unten angefügten Abbildungen erkennen.

Außerdem kann man erkennen, dass das Modell (bei korrekter Verwendung von Lüftungsstufe 1) bei den Daten von Filiale 1, nicht aber bei den Daten von Filiale 2 relativ genau ist. Ebenso kann man die starke Dynamik, hervorgerufen durch das Temperaturdifferenz-abhängige Ein- und Ausschalten erkennen. Nichtsdestotrotz muss auch dieses Modell wenn schon keiner Neuentwicklung, dann zumindest einer Anpassung an die Realität unterzogen werden.

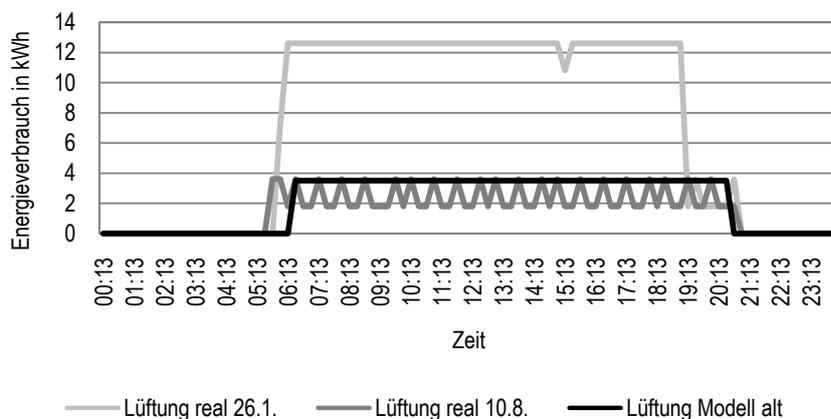


Abbildung 3.8: Modell und Realdaten einer Lüftungsanlage, Filiale 1

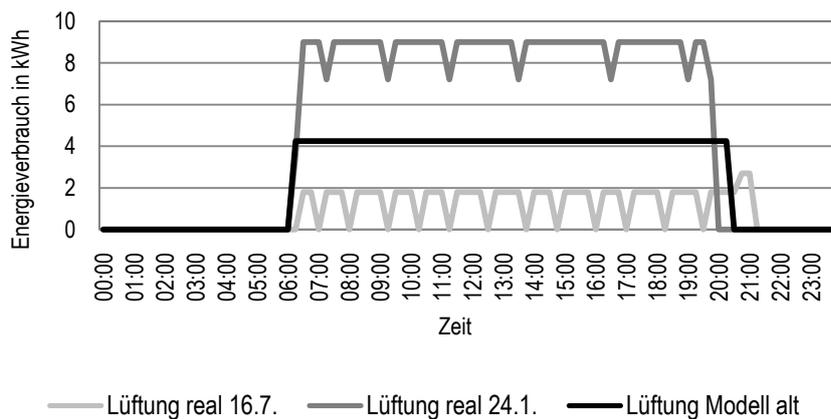


Abbildung 3.9: Modell und Realdaten einer Lüftungsanlage, Filiale 2

3.3 Neuentwicklung der Modelle

Wie im vorigen Abschnitt gezeigt wurde ist der Unterschied zwischen den realen Verbrauchsdaten einiger Energieverbraucher und den dazugehörigen Modellen ziemlich groß. Um bessere Ergebnisse zu erhalten ist es deswegen notwendig die Modelle zu verfeinern und neue zu entwickeln. Dabei soll das primäre Ziel sein die Modelle an die Realität anzunähern und genauer zu machen.

Dabei ist zu beachten, dass die Modelle nicht mehr so einfach sein müssen wie die alten, da die Analyse automatisiert werden soll und die Modelldaten nicht mehr per Hand in Tabellenkalkulationsprogramme oder ähnliches eingetragen werden müssen. Außerdem soll darauf geachtet werden, dass die Modelle sich nicht nur in ihren Stromverbrauchslevels den unterschiedlichen Geschäften und Typen von Geräten anpassen lassen, sondern auch bei einer Änderung der Geschäftsöffnungszeiten flexibel sind und an die neuen Gegebenheiten angepasst werden können.

Auch solche Modelle, bei denen eine Anpassung/Neuentwicklung nicht notwendig ist sollen soweit verallgemeinert werden, dass sie später mit allgemeinen Parametern beschrieben werden können.

3.3.1 Entwicklung eines Modells für Kälteanlagen

Als erstes wird das stark von den Realdaten abweichende Modell für Kälteanlagen, einer umfassenden Neuentwicklung unterzogen.

Bei den in Abbildung 3.6 und Abbildung 3.7 gezeigten Daten handelt es sich um Daten zweier Filialen, die beide um 07:30 Uhr aufsperrten und um 20:00 Uhr zusperren. Nun stellt sich die Frage weshalb der bereits erwähnte Mehrverbrauch auftritt. Nach Rücksprache mit Mitarbeitern von Envidatec stellt sich heraus, dass der erhöhte Stromverbrauch wahrscheinlich darauf zurückzuführen ist, dass die Mitarbeiter in den Filialen schon 2 Stunden vor den offiziellen Öffnungszeiten die isolierenden Rollos der Kühlregale, welche während der Nachtstunden verwendet werden, öffnen. Außerdem wird die Zeit, in der mit wenig bzw. gar keinem Kundenaufkommen zu rechnen ist, dafür genutzt um die Regale mit neugelieferten Waren aufzufüllen. Dieses Vorgehen belastet die Kühlaggregate

der Kälteanlagen sowohl durch die längere Zeit offenstehenden Türen als auch durch die größere Menge an zu kühlenden Waren.

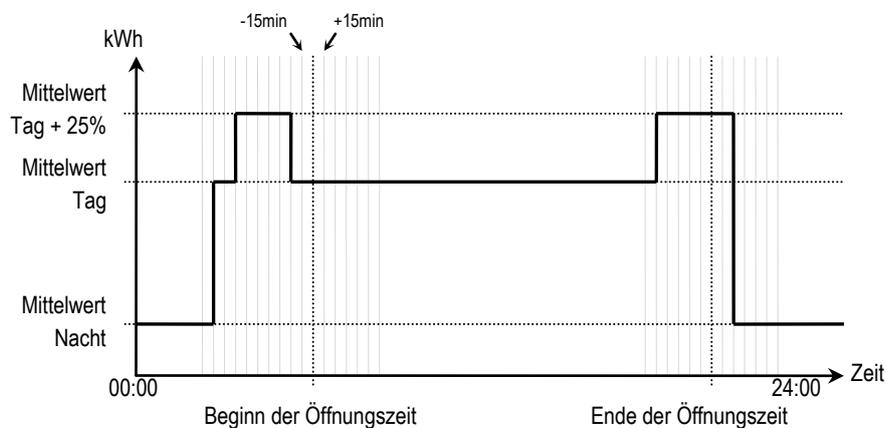


Abbildung 3.10: Neuentwickeltes Modell einer Kälteanlage ohne Temperaturoffset

Um das Modell nun zu verbessern liegt es nahe den nicht dem Nachtverbrauch entsprechenden Mehrverbrauch vor und nach den Öffnungszeiten hinzuzufügen. Außerdem fällt bei genauer Betrachtung auf, dass eine signifikante Erhöhung im Bereich des Beginns und des Endes der Öffnungszeiten auftritt. Nach Überprüfung aller vorhandener Filialen und verschiedener Tage im Verlauf des Jahres kann davon ausgegangen werden, dass in jeder Filiale eine solche Erhöhung von 20-30% des Tagesmittelwertes für 1,25 Stunden vor Beginn und 1,75 Stunden gegen Ende der Öffnungszeiten auftritt. Die erste Erhöhung kann zwischen 1,75 Stunden und 0,5 Stunde vor Beginn der Öffnungszeiten gemessen werden und die zweite zwischen 1,25 Stunden vor und 0,5 Stunde nach Ende der Öffnungszeiten.

Die gefundenen Ergebnisse ergeben ein vorläufiges Modell wie es in Abbildung 3.10 gezeigt wird. Dabei wurden die Erhöhungen mit einem Wert von 125% des Tagesmittelwertes angenommen.

Da es sich bei Kälteanlagen um Verbraucher handelt, welche, stärker als viele andere, durch thermische Prozesse beeinflusst werden, sollte nun auch untersucht werden ob der Stromverbrauch eines solchen Gerätes durch höhere Temperaturen beeinflusst wird. Dazu wurden die Daten der Filialen am heißesten Tag und am kältesten Tag des Jahres 2007 verglichen. In Abbildung 3.11 sieht man den Stromverbrauch der Kälteanlage einer Filiale gemessen am 24. Jänner und am 16. Juli 2007. Dabei kann man einen starken Unterschied zwischen dem mittleren Verbrauch während des Tages, aber auch einen merklichen Unterschied zwischen den Verbräuchen in der Nacht, erkennen.

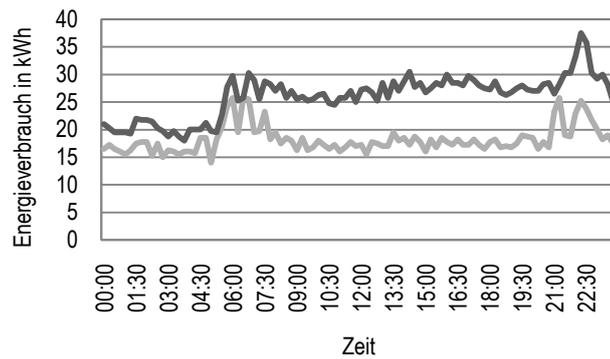


Abbildung 3.11: Vergleich des Stromverbrauchs einer Kälteanlage im Winter und Sommer

Auch bei den Daten aller anderen Filialen lässt sich solch ein Unterschied ausmachen. Um diesem Umstand Rechnung zu tragen wird das in Abbildung 3.10 vorgestellte Modell um eine Temperaturabhängigkeit erweitert. Um die Höhe eines einzuführenden Temperaturoffsets zu berechnen, wird folgende Überlegung angewendet: Für jedes Grad Celsius, welches die maximale Außentemperatur an einem Tag über dem Wert von 15 °C liegt, wird zu jedem Zeitpunkt ein fester Wert zum Modellwert des Stromverbrauchs hinzuaddiert. Bei einer Außentemperatur von 18 °C würde dieser Offset also dem dreifachen dieses Wertes entsprechen. Sollten keine Temperaturwerte vorhanden sein oder die maximale Außentemperatur kleiner als 15 °C sein, dann wird kein Offset zu den Modellwerten addiert. Um diese Annäherung zu erklären wurde eine Modellierung von wärmetechnischen Vorgängen durch elektrotechnische Grundlagen verwendet.

Dazu wird angenommen, dass eine Temperatur einer elektrischen Spannung und ein Wärmestrom einem elektrischen Strom entsprechen. In Abbildung 3.12 ist gezeigt wie ein elektrischer Wärmespeicher mit Hilfe dieses Modells dargestellt werden kann.

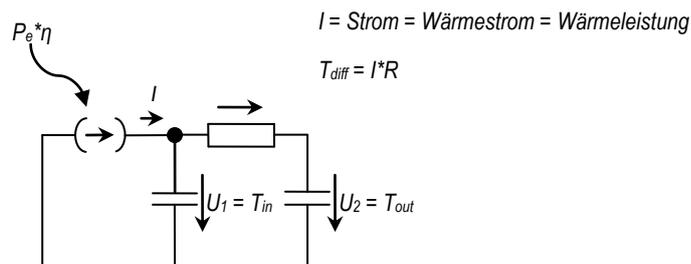


Abbildung 3.12: Modell eines elektrischen Wärmespeichers

Dabei stellen die Spannungen an den Kondensatoren die Temperatur innerhalb (U_1) und außerhalb (U_2) des Wärmespeichers dar. Der Widerstand kann als Isolierung oder Wärmewiderstand des Speichers interpretiert werden und der darüber fließende Strom als Wärmestrom. Die Multiplikation dieser beiden Werte ergibt die Temperaturdifferenz zwischen T_{in} und T_{out} .

Die Berechnung der einzelnen Werte kann dabei ebenso wie die Berechnung der elektrotechnischen Parallelwerte erfolgen, das bedeutet die Temperaturen (also die Spannungen) errechnen sich aus der Multiplikation von Wärmestrom mit dem Temperaturwiderstand.

Zur Aufrechterhaltung der im Wärmespeicher gespeicherten Energiemenge muss in der Realität konstant elektrische Energie hinzugefügt werden. Dies entspricht im Modell einer elektrischen Energie P_e , allerdings multipliziert mit einem Wirkungsgrad η , da Verluste bei der Energieumwandlung von elektrischer zu thermischer Energie auftreten. Um dieses Modell mit dem thermischen Modell zu verbinden wird angenommen, dass der Wärmestrom I gleich dieser Leistung P_e ist. Da U_1 konstant gehalten werden soll (in einem Kühlschrank soll z. B. immer die gleiche Temperatur herrschen), kann man den Strom in Kondensator 1 $I_1 = 0$ annehmen und daraus folgt dann $I = I_2$. Dadurch kommt man auf die folgende Formel:

$$U_1 = I * R + U_2$$

Und mit $P_e * \eta = I$ bzw. $U_1 = T_{in}$ und $U_2 = T_{out}$ folgt daraus:

$$P_e = \frac{T_{in} - T_{out}}{\eta * R}$$

Um die Linearität dieser Funktion sofort sichtbar zu machen, können zwei Konstante K_1 und K_2 verwendet werden. Die Gleichung sieht dann folgendermaßen aus:

$$P_e(T_{out}) = K_1 + T_{out} * K_2$$

Diese Ableitung unterstützt den gewählten Ansatz, das Temperaturverhalten mit Hilfe einer linearen Funktion anzunähern. Die Wahl der Werte für die beiden Konstanten wurde einerseits durch Analyse der vorliegenden Daten, andererseits durch eine Expertenmeinung abgesichert. Wobei noch einmal betont werden muss, dass diese Annäherung nur im Bereich des Arbeitspunktes von 15 °C angewandt werden kann.

3.3.2 Entwicklung eines Modells für Lüftungsanlagen

Wie bereits erwähnt handelt es sich bei Lüftungsanlagen um Geräte, welche stark an die Temperatur innerhalb und außerhalb des Geschäftes gekoppelt sind. Das Problem bei der Erstellung eines genauen Modells für die Lüftungsanlage ist allerdings, dass laut Aufgabenstellung zwar die Daten für die Außentemperatur vorliegen, nicht aber die Daten für die Innentemperatur.

Deswegen wurden Versuche unternommen eine Abhängigkeit des Stromverbrauchs einer Kälteanlage allein von der Außentemperatur abzuleiten. Doch das gelang nicht, da von der Außentemperatur nicht per se auf die Innentemperatur geschlossen werden kann, eine Kenntnis davon aber unbedingt notwendig ist, da die Regelung der Lüftung mit dieser Differenz gekoppelt ist. Man könnte sich das z. B. folgendermaßen vorstellen, dass falls diese Differenz einen bestimmten Sollbereich verlässt die Lüftung für eine bestimmte Zeit in einer bestimmten Stärke eingeschaltet wird. Wurde der Sollbereich wieder erreicht wird die Lüftung wieder abgeschaltet.

Um auf die Innentemperatur schließen zu können, wäre es notwendig eine Korrelation zwischen Außen- und Innentemperatur abzuleiten und genau solch eine Korrelation stellt die Lüftungsanlage, die modelliert werden soll, her. Womit klar ist, dass alleine mit einem dieser Werte, in diesem Fall

der Außentemperatur, nicht auf beide anderen Werte geschlossen werden kann. Hätte man allerdings einen zweiten Wert könnte man mit Hilfe dieses Wertes auf den dritten schließen. So wäre bei einer Kenntnis der Außen- wie der Innentemperatur sehr leicht ein Schluss auf den Stromverbrauch der Lüftungsanlage möglich, oder aus der Außentemperatur und dem Stromverbrauchs der Lüftungsanlage, sehr leicht die Innentemperatur abzuleiten. Alleine mit nur einem dieser Werte würde ein Schluss auf die beiden anderen allerdings nicht sehr fundiert sein.

Trotz dieses Problems ist es notwendig ein Modell für den Stromverbrauch einer Lüftungsanlage zu erstellen, da deren Stromverbrauch sich signifikant auf den Gesamtstromverbrauch niederschlägt. Aus dem bereits erwähnten Grund wird das erstellte Modell sehr einfach gehalten sein, da die genaue Abhängigkeit zwischen der Temperaturdifferenz und dem Modus in welchem die Lüftung operieren soll nicht modelliert werden kann. Um einen Idealverbrauch modellieren zu können muss deswegen auf das auch bisher verwendete Modell zurückgegriffen werden. Dieses Modell wurde bereits in Kapitel 3.2.2 vorgestellt. Dabei wird angenommen, dass die Lüftungsanlage im, relativ stromsparenden, Modus 1 einige Zeit vor dem Beginn der Öffnungszeiten eingeschaltet wird und am Ende der Öffnungszeiten ausgeschaltet wird.

Im Falle der Lüftungsanlage ist es außerdem von Interesse ob es sich beim zu analysierenden Tag um einen normalen Arbeitstag oder einen Sonntag gehandelt hat. Da an arbeitsfreien Tagen die Lüftung nicht aktiviert werden soll, sieht das Modell für solche Tage vor, dass die Lüftungsanlage im Idealfall keinen Strom verbraucht.

3.3.3 Entwicklung von Modellen für die Beleuchtung

Die Modelle für die Beleuchtungsstufen eines Geschäftes können, im Gegensatz zu anderen Modellen, sehr einfach erstellt werden. Der Grund dafür ist, dass eine Beleuchtungsanlage in der Zeit während der sie eingeschaltet ist konstant Strom verbraucht. Es gibt hier keine thermischen oder sonstigen Abhängigkeiten von anderen Prozessen. Auch saisonale Unterschiede durch die sich ändernde Länge der Tage sind hier zu vernachlässigen, da die Geschäfte im Winter wie im Sommer Beleuchtung benötigen.

Einzig zu beachten ist, dass meist zwei verschiedene Beleuchtungsmodi zur Verfügung stehen und verwendet werden. Da gibt es einerseits die sogenannte 1/3-Beleuchtung und andererseits die Vollbeleuchtung, bestehend aus 1/3-Beleuchtung (Beleuchtungsstufe 1) und 2/3-Beleuchtung (Beleuchtungsstufe 2).

Als grundsätzliche Vorgaben für die Beleuchtung haben sich folgende Einstellungen als Mittelweg zwischen energiesparendem Verhalten und dem grundsätzlichen Bedarf an ausreichender Beleuchtung bewährt: Die 1/3-Beleuchtung soll immer zwischen 1,5 Stunden vor Beginn und nach dem Ende der Öffnungszeiten eingeschaltet werden. In dieser Zeit befinden sich keine Kunden im Geschäft und das Licht reicht vollkommen aus um Neubestückung, Reinigung und ähnliches vorzunehmen. Die Vollbeleuchtung, also alle vorhandenen Beleuchtungsstufen zusammen, soll nur während der Öffnungszeiten zum Einsatz kommen. Meist wird das so gehandhabt, dass die Vollbeleuchtung 15 Minuten vor Beginn aktiviert und 15 Minuten nach Ende der Geschäftszeiten deaktiviert wird.

Zum Unterschied zum alten Modell, mit Hilfe dessen die gesamte Beleuchtung eines Geschäftes modelliert wurde, soll hier für jede der beiden Beleuchtungsstufen ein eigenes Modell erstellt werden. Diese Vorgehensweise ermöglicht zu einem späteren Zeitpunkt, dass auch andere Modelle für Beleuchtungsstufen, welche Abstufungen (z. B. 3 Beleuchtungsstufen) verwenden, ohne weiteres abgebildet werden können. In Abbildung 3.13 sind die beiden Modelle für die momentan bekannten Beleuchtungsstufen abgebildet.

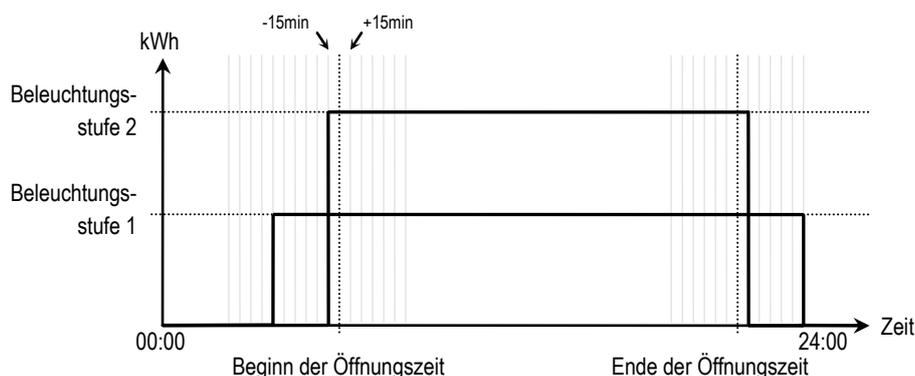


Abbildung 3.13: Neuentwickelte Modelle für verschiedene Beleuchtungsstufen

Das Modell für die Beleuchtung für Sonntage ist ähnlich einfach wie das für die Lüftungsanlage, da angenommen wird, dass die Beleuchtung an Tagen an denen der Betrieb geschlossen ist nicht eingeschaltet wird und deswegen keinen Strom verbraucht.

3.3.4 Entwicklung eines Modells für die Grundlast

Auch hier wird auf das bereits bestehende Modell für die Grundlast, welches in Abbildung 3.5 dargestellt ist, zurückgegriffen. Es ist deswegen keine Veränderung oder Neuentwicklung vorzunehmen, da sich diese Grundlast aus allen im Geschäft befindlichen Stromverbrauchern zusammensetzt, welche nicht durch eigene Modelle dargestellt werden können. Da es sich dabei um vollkommen unterschiedliche Verbraucher handelt und deswegen bewusst ein einfacher und allgemeiner Ansatz gewählt wurde um möglichst viele Eventualitäten abzudecken, kann kein genaueres Modell erstellt werden. Dadurch wird ein gewisser Fehler natürlich in Kauf genommen, welcher aber alleine dadurch aufgewogen wird, dass der Fehler einer Nichtberücksichtigung dieser Verbraucher einen noch schwerwiegenden Fehler hervorrufen würde.

Der Teilstromverlauf, welcher hier als Grundlast bezeichnet wird, setzt sich dabei aus den Stromverbräuchen von Geräten zusammen, die nicht in jeder der Filialen vorkommen und deswegen schwer zu verallgemeinern sind. Dabei wird angenommen, dass diese Geräte einen konstanten Stromverbrauch am Tag und einen konstanten Stromverlauf in der Nacht, welcher aber nicht dem Tagesstromverlauf entsprechen muss. Dadurch entsteht ein sehr einfaches Modell, welches zu Beginn der Öffnungszeiten einen Sprung vom Nachtstromverbrauch auf den Tagesstromverbrauch macht und am Ende der Öffnungszeiten wieder auf das Niveau des Nachtverbrauchs zurück.

Ein klarer Vorteil dieser Vereinfachung besteht darin, dass das Modell sehr vielfältig einsetzbar und nicht vom Vorhandensein bestimmter Verbraucher abhängig ist, dadurch können etwaige Besonderheiten und Unterschiede zwischen den Filialen ausgeglichen werden.

Da die Verbraucher, welche mit Hilfe der Grundlast zusammengefasst werden, nur innerhalb der Öffnungszeiten eingeschaltet sind, kann davon ausgegangen werden, dass an einem Sonntag derselbe Verbrauch wie in der Nacht vorliegt. Für den Fall, dass die Daten eines Sonntages analysiert werden sollen wird also angenommen, dass der Stromverbrauch während des Tages sich nicht von dem in der Nacht unterscheidet, also von 00:00 bis 24:00 dem mittleren Wert für den Nachtverbrauch entspricht.

3.4 Unterschied zwischen neuem und altem Modell

In diesem Abschnitt wird untersucht, wie sehr sich die modellierten Werte mit Hilfe der neuen Modelle verbessern lassen. Eine Verbesserung liegt dann vor, wenn die neuen Modellwerte näher an den realen Daten liegen als die alten Modellwerte. Um das zu überprüfen wird ein normierter mittlerer Fehler zwischen den Daten des neuen bzw. des alten Modells und den Realdaten eingeführt. Der normierte mittlere Fehler soll dabei definiert sein als:

$$Fehler_{normiert} = \frac{\sum_1^n |a(n) - b(n)|}{\sum_1^n b(n)}$$

$a(n)$ und $b(n)$ stellen in dieser Formel diskrete Funktionen dar. In diesem Fall handelt es sich bei diesen Funktionen um die Messwerte und Modellwerte, wobei $b(n)$ die Messwerte enthält und $a(n)$ die Modellwerte. Das Ergebnis der obigen Formel liegt zwischen 0 und 1 wobei der Fehler natürlich umso kleiner ist, je näher das Ergebnis an 0 liegt.

Hergeleitet wurde die Formel aus der bekannten Formel für den Fehler zwischen zwei Zahlen a und b , wobei nicht von Interesse ist, ob eine negative oder eine positive Abweichung zwischen den Zahlen vorliegt, weshalb der Absolutbetrag der Differenz gebildet wird. Aus allen Fehlern, welche zwischen den beiden Datenreihen auftreten wird nun ein mittlerer absoluter Fehler gebildet:

$$Mittlerer\ Absoluter\ Fehler = \frac{1}{n} * \sum_1^n |a(n) - b(n)|$$

Um nun eine Dimensionslosigkeit und Normierung dieses Wertes zu erreichen wird die eben vorgestellte Formel durch den Mittelwert der Realwerte dividiert um auf die zu allererst vorgestellte Formel zu gelangen.

$$Mittelwert\ der\ Realwerte = \frac{1}{n} * \sum_1^n b(n)$$

Diese Untersuchung wird sich auf das Modell einer Kälteanlage beschränken, da sich dieses Modell am auffälligsten vom alten Modell unterscheidet. Dazu werden zwei unterschiedliche Filialen ausgewählt und die Messdaten dieser Filialen an unterschiedlichen Tagen mit den Modelldaten des neuen und des alten Modells verglichen.

Die Messdaten umfassen dabei den Stromverbrauch der Kälteanlage, sowie die Außentemperatur an diesem Tag. Außerdem werden die entsprechenden Modelldaten ermittelt. Diese Daten werden dann einem MatLab/GNU Octave-Skript, welches für jedes der zwei Geschäfte zu diesem Zweck geschrieben wurde, übergeben und damit die normierten mittleren Fehler zwischen Modell und Realdaten ermittelt.

Bei solch einem Skript handelt es sich um eine Abfolge von Berechnungsbefehlen in der von MatLab und GNU Octave unterstützten Skriptsprache „m“. Die Befehle werden dabei in einer einfachen Textdatei welche mit der Dateierdung „.m“ versehen wird gespeichert und kann dann sehr einfach in einem der beiden Programme ausgeführt werden. Nähere Informationen zur Implementation in MatLab/GNU Octave sind in Kapitel 6 zu finden.

Wie man in der Gegenüberstellung der Fehler in Abbildung 3.14 sehen kann, ist der mittlere normierte Fehler zwischen neuem Modell und den Realdaten in allen Fällen kleiner als der Fehler zwischen altem Modell und den Realdaten. Herauszustreichen ist, dass besonders bei den Tagen, an denen der eingeführte Temperatur-Offset mitwirkt, eine starke Verbesserung zu bemerken ist.

Die Daten, welche diesen Rechenergebnissen zugrunde liegen, sind in Anhang C zu finden. Dabei handelt es sich um alle Messdaten der beiden Filialen, die von den betreffenden Tagen vorliegen, sowie den Werten für die berechneten Modelle.

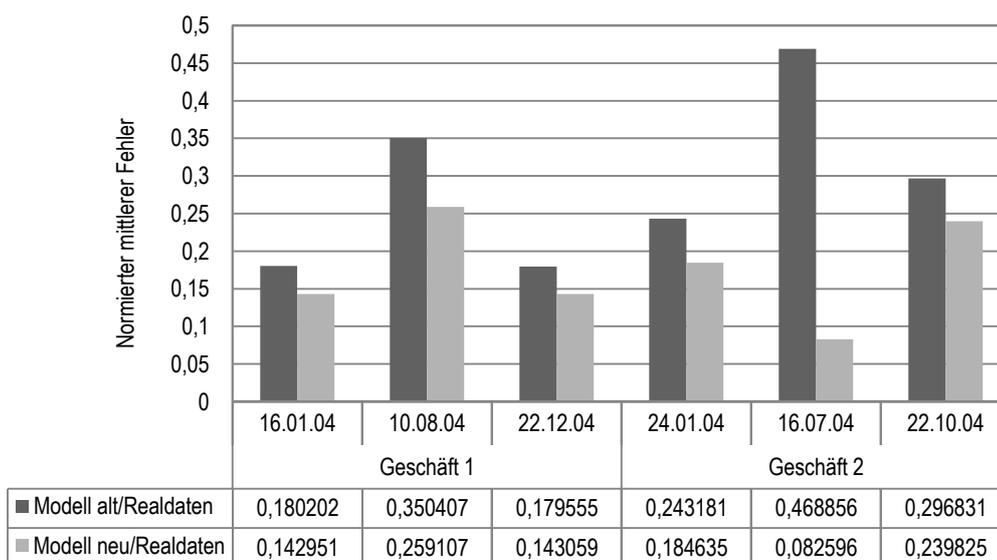


Abbildung 3.14: Vergleich der alten und der neuen Modelle mit dem Realverlauf

3.5 Parametrisierung der Modelle

Um die Modelle allgemein beschreiben zu können, unabhängig davon um welchen Stromverbraucher es sich handelt, muss ein allgemeines Parameterset definiert werden. Dieses Parameterset soll eine eindeutige Definition bei möglichst geringer Redundanz ermöglichen und alle entwickelten Modelle sollen dadurch dargestellt werden können.

Die folgenden Definitionsparameter wurden nach Analyse der in dieser Arbeit entwickelten Modelle als unbedingt notwendig erachtet:

- Anzahl der Messpunkte pro Stunde
- Beginn der Öffnungszeiten
- Ende der Öffnungszeiten
- mittlerer Stromverbrauch pro Tag (in kWh)
- mittlerer Stromverbrauch pro Nacht (in kWh)
- Typ des Stromverbrauchers
- zu verwendender Modelltyp
- allfällige zusätzliche Messwerte

Nachfolgend eine Erklärung warum die einzelnen Parameter ausgewählt wurden.

Der Parameter „Anzahl der Messpunkte“ wurde deshalb zusätzlich zu den sehr spezifischen Modelleigenschaften hinzugefügt um zu gewährleisten, dass die Modelle auch an eine mögliche spätere Vergrößerung der Anzahl der Messpunkte und eine damit verbundene Verkleinerung des Messintervalls angepasst werden können.

Die vier Parameter, welche die Öffnungszeiten und die Stromverbrauchslevels betreffen stellen die eigentlichen Betriebswerte (Dauer der Benutzung und Leistungswerte der Geräte) dar und sind unerlässlich für die Erstellung der Modelle.

Bei der genaueren Betrachtung des oben vorgestellten Parametersets könnte der Eindruck entstehen, dass die beiden Parameter „Typ des Stromverbrauchers“ und „Modelltyp“ weitgehend redundant sind, was allerdings nicht der Fall ist. Denn obwohl es möglich wäre das zu verwendende Modell vom Typ des Stromverbrauchers direkt abhängig zu machen, wurde ein eigener Parameter für die Identifikation des verwendeten Modells definiert um unterschiedliche Modelle für Stromverbraucher desselben Typs verwenden zu können. Dadurch wäre es beispielsweise möglich für Kälteanlagen, die einen gänzlich anderen Stromverbrauch als jene in Kapitel 3.3.1 aufweisen, neue Modelle zu definieren und trotzdem zu gewährleisten, dass diese Modelle als korrekte Modelle einer „Kälteanlage“ erkannt werden.

Der letzte Parameter „Allfällige zusätzliche Messwerte“ ist für Modelle wichtig, bei denen Abhängigkeiten mit anderen Prozessen, beispielsweise der Temperatur, bestehen. Bei den in den vorigen Kapiteln vorgestellten Modellen trifft das besonders auf das Modell der Kälteanlage zu, bei dem im Falle hoher Temperaturen ein Offset auftritt. Das muss sich aber nicht auf Temperaturmesswerte beschränken, beispielsweise könnten in anderen Fällen durchaus Korrelationen mit anderen Messwerten auftreten, wie z. B. Sonneneinstrahlung bei Jalousien.

Zum Abschluss kann man zusammenfassen, dass mit Hilfe dieser Parameter und der entwickelten Modelle jeder Verbraucher, der einem der bekannten Modelltypen entspricht, vollständig definiert und modelliert werden kann. Durch die Parametrisierung kann ein Verbraucher modelliert werden, ohne die speziellen Eigenschaften eines bestimmten Fabrikats kennen zu müssen. Die in den Kapiteln 3.2.3 - 3.2.2 vorgestellten Modelle sind durch die hier vorgestellten Parameter vollständig definierbar. Ein Parameterset für eine Kühlanlage könnte z. B. so aussehen: [4, „08:00“, „20:00“, 20, 10, „Kühlanlage“, „Kühlanlage Modell 1“, {2, 1, 2, ... 2, 1 }].

Mit der Verallgemeinerung der Modelle zum Abschluss dieses Kapitels ist es nun möglich sämtliche Energieverbraucher in einer sehr allgemeinen Form darzustellen und mit Hilfe der entwickelten Modelle den Stromverbrauchsverlauf eines solchen Verbrauchers für einen ganzen Tag zu modellieren. Das ist vor allem für die spätere Implementation von Vorteil, da dadurch alle Energieverbraucher auf die gleiche Weise behandelt und dargestellt werden können.

4. Bewertung von Energieverbrauchsdaten

Einer der Kernpunkte dieser Arbeit soll es sein, den gemessenen Stromverlauf eines Geschäftes oder einer Filiale einer größeren Filialkette zu analysieren und zu bewerten. Diese Bewertung soll einen Vergleich zwischen verschiedenen Filialen oder auch für die Daten einer Filiale an verschiedenen Tagen ermöglichen. Dabei ist es wichtig, dass diese Bewertung eindeutig und nachvollziehbar vorgenommen wird, und auch der Stromverbrauch von einer längeren Zeitperiode soll bewertet werden können.

Um einen geeigneten Bewertungsmodus für die vorliegende Problemstellung auswählen zu können, werden im ersten Teil dieses Kapitels verschiedene Ansätze solch eine Bewertung im Falle des Stromverbrauchs durchzuführen vorgestellt und auf ihre Aussagekraft überprüft. Dabei wird sowohl auf die bisher von der Firma Envidatec verwendet Benchmark als auch einige andere Ansätze eingegangen. Dabei ist herauszustreichen, dass für Energiemonitoring und –management Benchmarks bisher eher unbekannt und in der wissenschaftlichen Literatur nicht vertreten sind. Es soll dabei auch noch einmal genauer auf die Bewertung von Lastprofilen eingegangen werden, wie sie bereits ansatzweise in Kapitel 2.3 vorgestellt wurde.

Der zweite Teils dieses Kapitels widmet sich der Entwicklung einer aussagekräftigen neuen Benchmark. Mit Hilfe dieser Vergleichsgröße soll auf einen Blick eine Aussage über das Stromverbrauchsverhalten in einem Geschäft getroffen werden können, und mögliche starke Abweichungen vom Idealverbrauch sollen dadurch möglichst leicht sichtbar gemacht werden. Diese Benchmark hat dabei nicht den Anspruch, dass jene Objekt, die als „am besten“ eingestuft wurden auch wirklich in allen Bereichen am besten sind, es soll nur ausgesagt werden, dass das Optimierungspotential bei anderen Objekten größer ist.

Die zu entwickelnde Benchmark soll also ganz klar diejenigen Objekte herausstreichen helfen, bei denen das Optimierungspotential am größten ist. Das sind einerseits natürlich jene, wo auch der Energieverbrauch sehr hoch ist, auf der anderen Seite aber sollen die Geschäfte gefunden werden, die unter ihren Möglichkeiten bleiben, in dem Sinne als dass dort mehr Energie verbraucht wird, als eigentlich notwendig wäre.

4.1 Vorhandene Ansätze zum Vergleich von Energieverbrauchsdaten

Wie gut oder schlecht der Stromverbrauch eines Geschäftes, eines Büros oder aber auch eines Privathaushaltes zu bewerten ist, hängt sehr stark von den Prioritäten ab, die der Bewertung zu Grunde

gelegt werden. Während in so manchem Büro die Grundlast durch den permanenten (oft notwendigen) Betrieb von mehreren Personal Computern oder Servern sehr hoch ist, ist der Stromverbrauch solch eines Büros aber nicht von vornherein als schlecht einzustufen. Dieselbe Situation sollte allerdings in einem Privathaushalt eher vermieden werden.

Für die Bildung von Benchmarks zum Vergleich von Energieverbrauchsdaten gibt es zwar einige verschiedene Ansätze, aber keine wirklich allgemeine Vorgangsweise. So gibt zum Beispiel die Broschüre „Tipps zum Energiesparen für den Fleischer Mustermann“ der Stadt Hamburg aus dem Jahr 2004 an, dass der Gesamtstromverbrauch eines Jahres durch die eingekaufte Menge an Fleisch in Kilogramm zu dividieren ist um eine so genannte Energiekennzahl zu bekommen. Mit Hilfe dieser Energiekennzahl kann dann jeder Fleischer einschätzen, wie gut oder schlecht der Stromverbrauch eines Jahres im Gegensatz zu anderen Jahren war. Dabei gilt natürlich je besser das Verhältnis von Fleisch zu Energieverbrauch, also je kleiner der Quotient ist, desto besser war das Energieverbrauchsverhalten.

Eine Aussage über die wirkliche Qualität des Stromverbrauches lässt sich aus solchen Vergleichswerten aber nicht treffen, inkludiert dieser Ansatz doch, dass ein höherer Stromverbrauch bei gleichzeitig höheren Verkaufszahlen tolerierbar ist. Außerdem ist auch in dieser Broschüre angefügt, dass es sich dabei um „eine tendenzielle Aussage“ handelt, da „die unterschiedliche Anlagenausstattung sowie das ungleiche Warensortiment verschiedener Betriebe eine direkte Vergleichbarkeit verhindern“. [HAM04]

Ein ganz ähnlicher Ansatz zur Berechnung einer Benchmark zur Bewertung des Energieverbrauchs wurde bisher auch von der Fima Envidatec verwendet. Die Benchmarks wurden hier bisher folgendermaßen berechnet: Es wurde der Gesamtstromverbrauch eines ganzen Jahres durch Addition sämtlicher Messwerte gebildet und die erhaltene Summe durch die Quadratmeteranzahl des Geschäftes dividiert. Dadurch konnte der Gesamtstromverbrauch verschiedener Geschäfte untereinander verglichen werden.

Auch diese Benchmark hat ihre Nachteile. So ist hier die wichtigste Einflussgröße der Gesamtjahresstromverbrauch, denn es ist anzunehmen, dass sich die Quadratmeteranzahl eines Geschäftes über die Jahre hinweg nicht ändert. Der Vorteil solch einer Vorgehensweise ist, dass der Stromverbrauch auf die Geschäftsgröße normiert wird. Was allerdings total vernachlässigt wird sind Einflussfaktoren wie die Bausubstanz der Gebäude und die darin befindlichen Geräte. Oft sind aus baulichen Maßnahmen, Kriterien vorgegeben, die keinen geringeren Energieverbrauch ermöglichen. So könnte ein Geschäft innerhalb seiner Möglichkeiten einen beinahe optimalen Stromverbrauch aufweisen und doch bei der Bewertung durchfallen.

In Abbildung 4.1 ist der Tagesstromverbrauch eines Geschäftes dargestellt. Bei der eben vorgestellten Art solch einen Tagesverlauf zu benchmarken würden die Werte aller Messpunkte zusammengerechnet und danach durch einen bei jeder Filiale unterschiedlichen Faktor (in unserem Beispiel der Filialgröße) dividiert. Die vorliegende Abbildung ist allerdings nur der Tagesverlauf, in der verwendeten Methode würden die Einzelwerte von 365 beziehungsweise 366 solcher Verläufe zusammengerechnet, da es sich dabei wie bereits beschrieben um eine Jahresbenchmark, also den Energieverbrauch eines Jahres gehandelt hat.

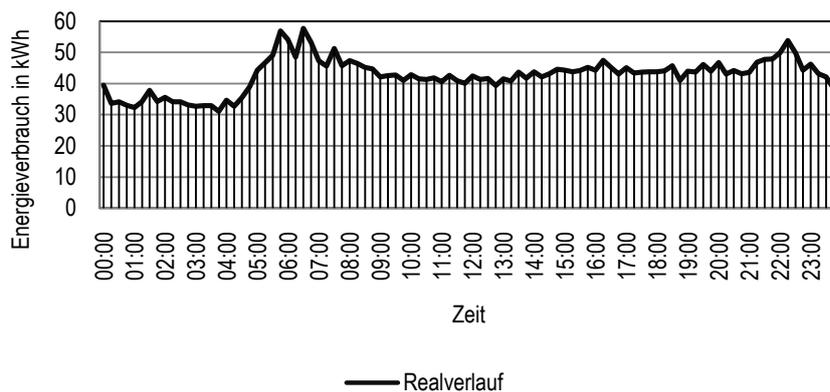


Abbildung 4.1: Tagesenergieverbrauch einer Filiale

Auf den ersten Blick ist aus diesen Ansätzen ersichtlich, dass eine Einteilung der zu bewertenden Objekte in verschiedene Klassen, sei es durch Normierung auf die Objektgröße (in m²) oder die verkaufte Fleischmenge, eine erfolgsversprechende Strategie ist. Damit ist gewährleistet, dass nur vergleichbare Objekte wirklich verglichen werden, da ein Vergleich zwischen einer Filiale eines Supermarkts und einem Privathaushalt weder repräsentativ, noch wirklich aussagekräftig ist. Auch gibt es, gerade bei Supermarktketten, signifikante Unterschiede in der Größe und Ausstattung der Filialen, was eine solche Einteilung ebenso rechtfertigt.

Doch auch bei solch einer geeigneten Clusterung der Objekte führt der Weg über den Jahresgesamtstromverbrauch in eine Sackgasse, da man dadurch zwar einen Vergleich zwischen den Werten einzelner Jahre ziehen kann, aber man nie sagen kann ob ein Wert, rein objektiv betrachtet, ein gutes oder schlechtes Stromverhaltensverhalten zum Ursprung hat. Hier könnte der Einwand, dass ein niedriger Gesamtwert, hervorgerufen durch einen niedrigen Stromverbrauch, objektiv besser zu beurteilen ist einige Berechtigung haben. Trotzdem kann es sein, dass bei solch einer Beurteilung Geschäfte/Filialen etc. schlecht beurteilt werden, obwohl sie im Rahmen ihrer Möglichkeiten ein sehr positives Verbrauchsverhalten haben, aber durch Umstände wie veraltete Bausubstanz oder besonders verbrauchsintensive Geräte einen höheren Grundverbrauch haben, als ein vergleichbares Objekt.

Was also sowohl beim ersten, wie auch beim zweiten Ansatz, also der Normierung auf eine Fleischmenge oder die Baugröße, fehlt ist die geeignete Vergleichsbasis, die festlegt ab wann es sich um ein schlechtes bzw. gutes Verbrauchsverhalten handelt.

Eine mögliche Lösung für dieses Problem ist es die Benchmark des Prozesses oder Objektes im idealen Anwendungsfall zu berechnen und dann mit der aus realen Daten gewonnenen Vergleichszahl verglichen werden kann. Man würde also eine „ideale“ Filiale modellieren und an Hand dieser Filiale vergleichen wie gut oder schlecht der Stromverbrauch war.

Was für viele Fälle die geeignete Lösung darstellt, verlagert hier das Problem allein auf die Frage welches der Geschäfte als „ideales“ Geschäft angesehen wird und was für dieses ideale Geschäft ein „idealer“ Tagesverbrauchsverlauf ist.

4.2 Entwicklung einer aussagekräftigen Benchmark

Wie in den ersten Teilen dieses Kapitels ausgeführt soll eine neue, aussagekräftige, Benchmark entwickelt werden. Mit Hilfe dieser Kennzahl, denn um nichts anderes handelt es sich bei einer Benchmark soll der gemessene Stromverlauf eines Geschäftes, oder allgemein jedes Objektes, bewertet werden können. Diese Bewertung soll sich in einer Zahl ausdrücken, mit deren Hilfe das zu untersuchende Objekt mit anderen Objekten derselben Art oder desselben Typs verglichen werden kann. Außerdem soll es diese Kennzahl ermöglichen, dass der Verbrauch eines Objektes direkt bewertet werden kann.

Ein bereits angesprochener Lösungsansatz, der genauer untersucht werden soll ist jener, anzunehmen, dass es einen idealen, weil minimalen, Verbrauch für jedes Geschäft zu jeder Zeit gibt. Eine Idee wäre nun ein ideales Geschäft zu erstellen und jedes der Geschäfte damit zu vergleichen. Durch das bereits beschriebene Problem der Clusterung ist aber auch dieser Ansatz nicht geeignet um eine befriedigende Lösung zu liefern. Da gewährleistet werden müsste, dass das besprochene ideale Geschäft sowohl für sehr große, als auch für sehr kleine Geschäfte eine berechnete Vergleichsbasis darstellt.

Was aber wäre, wenn es möglich wäre einen Idealverlauf für jedes einzelne Geschäft zu generieren, den Realverlauf damit zu vergleichen und aus diesem Vergleich auf die Qualität des Stromverbrauchsverhaltens schließen zu können? Genau dieser Ansatz soll hier verfolgt werden und gezeigt werden, dass mit der bereits erfolgten Vorarbeit dieser Ansatz zielführend sein kann.

Um ein Idealmodell für den Tagesverlauf eines Geschäftes zu erstellen müssen einige Parameter bekannt sein. Grundlage bilden die Informationen über die in diesem Geschäft vorhandenen Verbraucher und Messdaten betreffend den Gesamtstromverbrauch, also das Lastprofil, und die Außentemperatur. Letzteres ist vor allem wichtig um die Reaktion und das Verhalten thermisch stark beeinflussbarer Verbraucher richtig einschätzen zu können.

Wenn sämtliche Informationen über Verbraucher und Geschäft vorliegen kann für jeden Tag, des zu bewertenden Zeitraumes, ein ideales Gesamtmodell für das Lastmodell erstellt werden. Dazu wird für jeden der Verbraucher und jeden Tag ein separates Modell erstellt. Ein Modell pro Verbraucher für alle Tage kann deswegen nicht ausreichen, da sich die Modelle einzelne Verbraucher an verschiedenen Tagen, durch Schwankungen der Außentemperatur, sehr von einander unterscheiden können.

Aus den Modellen der Verbraucher für einen Tag kann man nun das gewünschte Gesamtmodell für diesen Tag erstellen. Dieses Gesamtmodell besteht aus der Summe aller Modelle bekannter Verbraucher für diesen Tag. Diese Modelle stellen den idealen, also minimalen, Verbrauch für den von ihnen repräsentierten Verbraucher dar. Durch die Summenbildung entsteht also ein Modell des minimal möglichen Verbrauches eines Geschäftes an diesem Tag, abhängig von der Außentemperatur und unter der Annahme, dass alle Verbraucher ihrem Idealmodell entsprechend betrieben wurden.

Dieses Gesamtmodell wird nun mit den realen Messwerten, welche vom zugehörigen Tag vorliegen verglichen. Und zwar wird der Unterschied zwischen den beiden Verläufen berechnet, indem man

die Modellwerte zu jedem Zeitpunkt des Tages von den Messwerten subtrahiert. Dadurch entsteht ein Zeitverlauf der Abweichungen der Realdaten vom Idealmodell. Es ist daraus herauszulesen um wie viel der Stromverbrauch zu jedem beliebigen Zeitpunkt eines Tages vom Idealzustand abgewichen ist. In

Abbildung 4.2 ist der Unterschied zwischen dem Tages-Gesamtmodell und dem Realverlauf gut zuerkennen. Die schwarz schraffierte Fläche in dieser Grafik stellt die Differenzwerte aller vom idealen Modellverlauf abweichenden Werte dar.

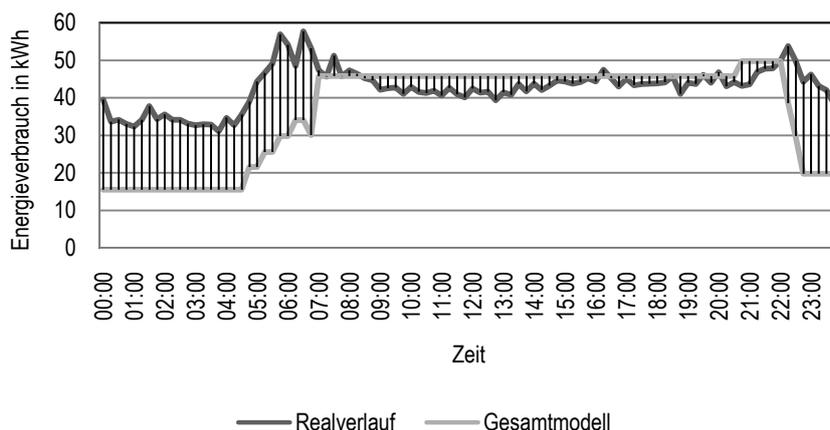


Abbildung 4.2: Unterschied zwischen Realverlauf und Tages-Gesamtmodell

Aus diesen Abweichungen wird dann eine Summe gebildet, welche die gesamte Abweichung vom Idealverlauf an einem Tag beinhaltet. Sollten mehrere Tage bewertet werden, werden die Summen der Abweichungen dieser Tage noch einmal zu einer Gesamtsumme aufsummiert. Man könnte diese Summe auch als Integral über die Differenzfläche verstehen. Die folgende Formel stellt diese Berechnung mathematisch dar.

$$Benchmark = \sum_1^t (E[n] - M[n])$$

Hier stellen die Funktionen $E[n]$ die Messdaten des Stromverbrauchs und $M[n]$ die Modellwerte für ebendiesen dar. Bei beiden Funktionen handelt es sich um diskrete Funktionen, weswegen in der Formel eine Summe statt einem Integral Verwendung findet. Der Wert t steht für alle Messzeitpunkte des Intervalls, dessen Stromverbrauch bewertet werden soll.

Die so entstandene Gesamtsumme kann nun folgendermaßen interpretiert werden: Diese Summe stellt den Unterschied zwischen dem idealen Lastprofil und den realen Daten dar. Wäre diese Summe gleich 0 würden die realen Daten dem Ideal entsprechen und das Stromverhaltensverhalten könnte als „gut“ bewertet werden. Je größer aber die Summe ist, desto größer ist auch die Abweichung von Ideal und desto mehr Strom wurde unnötigerweise verbraucht. Je größer also diese Gesamtsumme ist, desto schlechter ist der Stromverlauf zu bewerten.

Man könnte diese Gesamtsumme auch das mögliche „Einsparungspotential“ in diesem Geschäft/Betrieb/Objekt nennen, denn es entspricht dem Strom, der nicht notwendigerweise verbraucht wurde und eigentlich eingespart werden könnte oder besser sollte.

Bei dieser Betrachtungsweise ist es gleichgültig wie hoch der tatsächliche Stromverlauf in den einzelnen Geschäften ist, denn es wird nur die Abweichung vom Ideal betrachtet. Das hat den Vorteil, dass auch Filialen welche durch veraltete Bausubstanz, alte Geräte oder andere externe Umstände einen erhöhten Energieverbrauch haben durch aktives Energiemanagement eine gute Benchmark erreichen können.

Durch die eben vorgestellte Benchmark ist es auch möglich das zu untersuchende Zeitintervall frei zu wählen. Dadurch, dass für diese Benchmark die Tage einzeln analysiert werden und die an den einzelnen Tagen erreichten Benchmarks aufaddiert werden ist es unerheblich ob das Intervall einen Tag oder ein halbes Jahr beträgt. Das hat den Effekt, dass vollkommen frei gewählt werden kann, wie oft die Benchmark berechnet wird. Auf diese Art und Weise können auch einzelnen Tage mit stark abweichendem Energieverhaltensverhalten sehr leicht entdeckt werden, da die Benchmark an diesen Tagen natürlich viel schlechter ist als an anderen. Am Ende eines längeren Intervalls kann trotzdem eine Gesamtbenchmark, bestehend aus den Tagesbenchmarks berechnet werden, welche denselben Wert hat wie wenn man auf die Tagesbenchmarks verzichtet hätte und nur dieses längere Intervall untersucht hätte. Dadurch sind individuell konfigurierte Berechnungsintervalle möglich ohne das Gesamtergebnis dadurch zu verfälschen.

In der Tabelle 2 sind beispielhaft die Einsparungspotentiale (in kWh), also die Benchmarks, einer Filiale an zwei Tagen des Jahres 2007. Wie leicht zu erkennen ist, ist das Stromverhaltensverhalten am ersten der beiden Tage (24. Jänner 2007) eindeutig besser einzustufen. Bei der durchgeführten Berechnung wurden natürlich auch die in diesen Geschäften installierten Geräte eruiert und die entsprechenden Modelle aus den Grundwerten dieser Geräte berechnet.

Tabelle 2: Einsparungspotentiale einer Filiale an zwei Tagen im Jahr 2007

Datum	24.01.2007	05.12.2007
Benchmark (kWh)	184,7	570,18

Natürlich kann auch solch eine Benchmark nicht die genaue Analyse des Stromverhaltensverhaltens für einzelne Filialen ersetzen. So sollte auch weiterhin der Gesamtverbrauch im Auge behalten werden. Allerdings kann durch die Verwendung solch einer Benchmark sehr gut herausgefunden werden bei welchem der zu bewertenden Objekte sich die Analyse am meisten lohnen könnte, da bei diesen Objekten die größte Einsparung möglich wäre.

4.3 Mögliche Erweiterung der neuen Benchmark

Wie es in solchen Fällen immer der Fall ist gäbe es auch bei der im vorigen Kapitel vorgestellten Benchmark einige Möglichkeiten der Steigerung der Effizienz. So kann zwar mit Hilfe der im vorigen Abschnitt entwickelten Benchmark das Stromverhaltensverhalten in einem Geschäft bewertet

werden, da sie das Einsparungspotential dieser Filiale, abhängig von den filialspezifischen Parametern, darstellt. Allerdings könnte man hier einwerfen, dass bei größeren Filialen, mit mehr Verbrauchern, natürlich auch das Einsparungspotential größer ist. Es wäre also durchaus zu diskutieren ob man zusätzlich noch eine Anpassung der Benchmark auf die Größe des Geschäftes, also eine Art Normierung, durchführt.

So könnte man beispielsweise die Benchmark in der Form verfeinern, dass man den berechneten Wert auf die Größe der Filiale normiert. Das könnte auf verschiedenste Art und Weise bewerkstelligt werden. Zum Beispiel indem die Benchmark durch die Filialgröße in m^2 dividiert. Oder aber man dividiert die Benchmark durch den Gesamtstromverbrauch, der im zu untersuchenden Zeitraum aufgetreten ist.

Erstere Möglichkeit würde alleine die Größe der Filiale in Betracht ziehen, was allerdings wieder eine Einschränkung darstellt. Eine Einschränkung ist das deswegen, weil sich Filialen flächenmäßig durchaus sehr ähnlich sein können aber in Bezug auf die installierten Stromverbraucher fundamental unterscheiden können.

So gesehen stellt die zweite Möglichkeit eine eindeutige Verbesserung dar. Durch die Division des Einsparungspotentials durch den Gesamtstromverbrauch würde das absolute Einsparungspotenzial einer Filiale in direkten Bezug zur gesamten verbrauchten Energiemenge gesetzt und man würde dadurch ein relatives Einsparungspotential erhalten. Deswegen würde die zweite Möglichkeit hier eindeutig eine Verbesserung zur entwickelten Benchmark darstellen.

Diese Möglichkeiten stellen natürlich nur weitere Verfeinerungen der bereits beträchtlich gesteigerten Qualität der Benchmark dar. Die Normierung auf Fläche oder Gesamtstromverbrauch wurde nur deswegen nicht fix in die neue Benchmark integriert, da dadurch die im vorigen Abschnitt besprochenen Möglichkeit, dass die Benchmarks einzelner Tage sehr leicht zu einer Gesamtbenchmark addiert werden können, verloren gehen würde. Die Grundaussage, dass die Benchmark das gesamte Einsparungspotential einer Filiale darstellt würde dadurch nichtmehr gegeben sein.

Allerdings könnte man die angesprochenen Normierungen trotzdem in einer erweiterten Analyse umsetzen.

Außerdem möglich wäre die multiplikative Verknüpfung der Benchmark, also des Einsparungspotentials mit dem im zu untersuchenden Intervall aufgetretenen mittleren Strompreis. Dadurch hätte man eine sehr plakative Möglichkeit zu zeigen wie hoch die Einsparungen wirkliche gewesen wären.

All diese Möglichkeiten einen Energieverbrauch zu bewerten, ersetzen natürlich niemals die genaue Untersuchung des Verbrauchsverhaltens und können immer nur unterstützend wirken.

5. Erkennung einzelner Energieverbraucher

Hier soll die mathematische Erkennung von Energieverbrauchern verschiedener Typen aus dem Lastprofil behandelt werden. Eine solche, rein mathematische, Erkennung ist deswegen von Interesse, da eine messtechnische Lösung für diese Aufgabenstellung nicht praktikabel ist. Das ist deswegen der Fall, da die Installation der Geräte, die eine solche Erkennung auf rein messtechnischem Weg ermöglichen würden, sehr hohe Kosten in Anschaffung und vor allem Wartung bedeuten würden.

In diesem Kapitel wird zuerst auf die Problematik der Erkennung von Energieverbrauchern eingegangen und verschiedene Möglichkeiten eine solche Erkennung durchzuführen vorgestellt. Danach wird auf die, im Rahmen dieser Arbeit, gefundene Lösung für das Problem eingegangen und warum diese den anderen Möglichkeiten vorgezogen wurde. Dabei wird gezeigt wie man die erhaltenen Lösungen durch die Zuhilfenahme eines weiteren, bereits vorgestellten, Ansatzes noch verbessern kann. Zuletzt wird noch auf mögliche Verbesserungen und Adaptierungen für die Zukunft eingegangen.

5.1 Problematik der Erkennung

Grundsätzlich handelt es sich bei einem Lastprofil um die Verbrauchsfunktion eines Geschäftes oder Verbrauchers über der Zeit. Man könnte hier also sehr einfach eine Parallele zur Signaltheorie herstellen, bei der zeitabhängige Funktionen, so genannte Signale, analysiert und verwendet werden. Wenn man nun das Lastprofil eines Verbrauchers als Signal sieht und mit dem Lastprofil eines anderen Verbrauchers addiert erhält man ein zusammengesetztes (Summen-)Signal beider Verbraucher. Führt man diesen Schritt für alle Verbraucher eines Geschäftes durch erhält man das Gesamtsignal (oder Lastprofil) dieses Geschäftes.

Wenn man das nun Lastprofil als Überlagerung verschiedener Signale betrachtet kann man sehr leicht erkennen, wo die Probleme liegen eines der Teilsignale zu detektieren, das ist vor allem der Fall wenn es sich nicht nur um 2 Teilsignale sondern um mehrere handelt.

Grundlegenden Probleme stellen natürlich, wie bei jedem vergleichbaren Verfahren, die möglichen Unsicherheiten im messtechnischen Bereich dar. Dabei kann es sich um das unvermeidliche Rauschen handeln, oder auch zu geringe Messauflösung. Es soll angenommen werden, dass man in diesem Fall solche Probleme weitgehend im Griff hat.

Des weiteren stellt die Erkennung eines einzelnen Signals aus einer großen Anzahl von überlagerten Signalen eine nicht zu unterschätzende Herausforderung dar. Das trifft aber nicht alleine nur auf die Interpretation von Lastprofilen als Signal, sondern für die gesamte Signaltheorie zu.

Zu guter Letzt ist nicht allein das einwandfreie Erkennen eines Teilsignals aus einem Gesamtsignal sehr schwierig, sondern es stellt eine beinahe noch größere Herausforderung dar eindeutig zu erkennen, dass ein Signal nicht Teil des Gesamtsignals ist. Der Grund hierfür ist, dass zuerst alle möglichen Fehler- und Störungsquellen ausgeschlossen sein müssen, und alle anderen vorhandenen Signale einwandfrei erkannt worden sein müssen, bevor der Schluss zulässig ist, dass ein bestimmtes Teilsignal nicht vorhanden ist.

5.2 Verschiedene Möglichkeiten zur Erkennung

Die Betrachtung der Verbraucher und des gesamten Lastprofils als Signale lässt für die Analyse derselben sofort an klassische Werkzeuge der Signaltheorie wie Fourieranalyse, Korrelation oder Pattern-Matching-Algorithmen denken. Auch Ansätze aus der Fuzzy-Logic oder aus dem Gebiet der sogenannten selbstlernenden Algorithmen könnten zielführend sein. In diesem Abschnitt sollen all diese Ansätze beschrieben werden und warum sie alle bei der vorliegenden Problemstellung nicht als Lösung in Frage kommen.

5.2.1 Fourieranalyse

Die Fourieranalyse stellt in der Signal- und Systemtheorie ein mächtiges Instrument zur Berechnung und Manipulation von Signalen dar. Besonders für Detektion von periodischen Signalen ist sie sehr gut anwendbar. Nähere Informationen zu Zeitreihenanalysen im Frequenzbereich sind in [FLA99] zu finden.

Um nun Herauszufinden ob solch eine Vorgehensweise im vorliegenden Fall zum Ziel führt, müssen die in Frage kommenden Signale im Verlauf eines Tages angesehen werden. Dazu wird hier noch einmal auf die Modelle aus Kapitel 2 hingewiesen. Diese Modelle bieten einen guten Eindruck wie die zu detektierenden Signale aufgebaut sind. Untersucht man diese Signale näher, kann man feststellen, dass keines davon sich in einem großen Maße von den anderen unterscheidet, sind es doch alle aperiodische Rechteckfunktionen, welche alle ungefähr zum selben Zeitpunkt einen Wechsel von einem stabilen Niveau zu einem anderen stabilen Niveau aufweisen und (ebenfalls wieder zum selben Zeitpunkt) einen weiteren Wechsel vom höheren Niveau zurück auf das niedrigere vollführen.

Durch diese sehr ausgeprägte Rechteckform bei den zeitlichen Signalen lassen sich bei einer Übertragung in den Frequenz-Bereich keine Peaks bei den charakteristischen Frequenzen erkennen, da die Repräsentation all dieser Funktionen eine (mehr oder weniger ausgeprägte) Funktion der Form $\frac{\sin(x)}{x}$ darstellt.

Hinzu kommt, dass auch der Realverlauf des Lastprofils die Merkmale einer Rechteckfunktion aufweist, also auch daraus nicht viel herausgelesen werden kann. Dieser letzte Punkt ist natürlich vor allem darin begründet, dass alle Teilsignale auch Rechteckfunktionen sind. Ein kleines Beispiel für diese Ausführungen ist in Abbildung 5.1 zu sehen.

Es handelt sich dabei um die Darstellung eines Realverlaufes eines ganz normalen Geschäftes und dessen Kühlanlage. Außerdem wurden die Daten des in Kapitel 2 entwickelten Modells für diese Kühlanlage dargestellt. Es ist gut zu erkennen, dass es (außer dem Peak bei der Frequenz 0) keine charakteristischen Frequenzpeaks gibt.

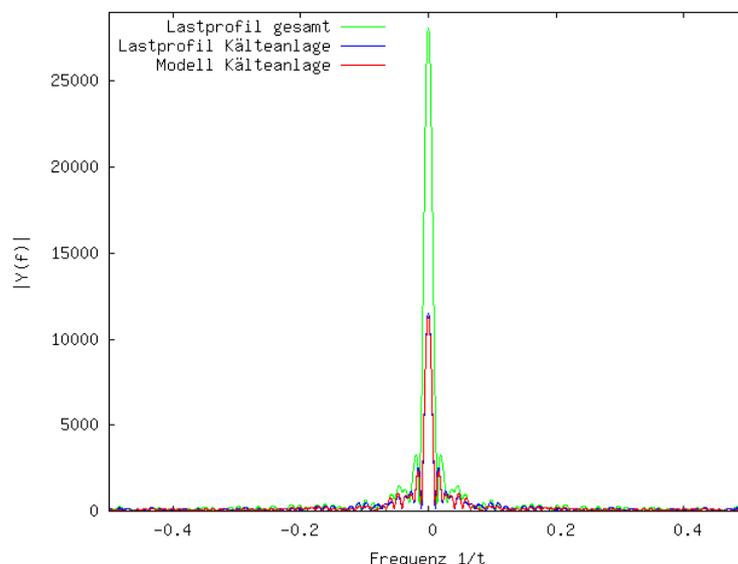


Abbildung 5.1: Signale im Fourierbereich

Um wirkliche Periodizität herzustellen könnte auch der Signalverlauf einer Woche analysiert werden, allerdings kann man daraus im allerbesten Fall herauslesen, ob ein bestimmter Verbraucher in dieser Woche gelaufen ist oder nicht. Davon aber auf einen speziellen Tag zu schließen, was eigentlich das Ziel der Bemühungen ist, ist allerdings beim besten Willen nicht möglich.

Aus diesen Gründen ist zu erkennen, dass die Fourieranalyse beim vorliegenden Problem nicht zur Lösungsfindung beitragen kann.

5.2.2 Korrelationskoeffizient zweier Signale

Unter dem Korrelationskoeffizient zweier diskreter oder kontinuierlicher Signale versteht man im Allgemeinen einen dimensionslosen Wert zwischen -1 und +1, der Auskunft über den linearen Zusammenhang zwischen den beiden Signalen gibt.

Stimmt der Wert mit einem der beiden Extrema (-1 oder +1) überein, besteht zwischen den Signalen ein vollständiger negativer oder positiver Zusammenhang. Nimmt der Korrelationskoeffizient den Wert 0 an, besteht kein linearer Zusammenhang. Allerdings schließt der letzte Fall nicht aus, dass zwischen den Signalen ein nichtlinearer Zusammenhang besteht. Der Korrelationskoeffizient lässt sich laut [BAM02] über folgende Formel berechnen:

$$CorCoef = \frac{E\{[x - E(x)][y - E(y)]\}}{\sigma_x * \sigma_y}$$

In dieser Formel stellen x und y die beiden Signale, E den Operator für den Erwartungswert und σ die Standardabweichung der Signale dar.

Der Korrelationskoeffizient stellt, wie bereits erwähnt, nur den linearen Zusammenhang zwischen zwei Signalen dar und dadurch haben z. B. die beiden konstanten Signale $f_1(t) = 1$ und $f_2(t) = 5$ trotz verschiedener Signalhöhen einen Korrelationskoeffizienten von 1.

Um nun in einem zusammengesetzten Gesamtsignal ein Teilsignal einwandfrei detektieren zu können, kann nicht so vorgegangen werden, dass zuerst all jene Signale, die nicht von Interesse sind, und etwaige Störsignale aus dem Gesamtsignalsignal heraus gerechnet werden um danach den Korrelationskoeffizienten zwischen einem Modell des gesuchten Signals und dem verbleibenden Rest zu bilden. Hier stellt sich, neben dem Problem der Aufbereitung des Gesamtsignals auch jenes, dass selbst wenn so eine Aufbereitung möglich wäre nur den linearen Zusammenhang zwischen dem Restsignal und dem Modell berechnen.

Und genau aus diesem Zusammenhang lässt sich nicht einwandfrei schließen ob das gesuchte Signal nun aufgetreten ist oder nicht, da sich die äußeren Formen der Signale (und ihrer Modelle) zu ähnlich ist. Man stößt hier also auf ein ähnliches Problem wie im vorherigen Abschnitt über die Fourieranalyse, nur eben nicht im Frequenz- sondern im Zeitbereich.

5.2.3 Pattern Matching

Unter Pattern-Matching-Verfahren versteht man Algorithmen welche ein gegebenes Muster in einem eingeschränkten Suchbereich finden sollen. Um solch ein Verfahren auf Signale anzuwenden, müssen diese Signale zuerst aufbereitet werden. Dazu werden alle bekannten Fremdsignale und alle vermuteten Störsignale aus dem Grundsignal gefiltert und das verbleibende aufbereitete Signal einem Patter-Matching-Verfahren unterzogen. Dabei wird die Form und Größe des Signals mit einem Mustersignal verglichen. Für weitere Informationen zum Thema Pattern-Matching wird hier auf [HAN02] verwiesen.

Im vorliegenden Fall stellt sich (wie bereits im Abschnitt über den Korrelationskoeffizienten erwähnt) das Problem des zu geringen Kenntnisstandes über etwaige andere Signale. Das bedeutet, dass man nicht davon ausgehen kann, dass bestimmte Signale immer und andere nie auftreten, da es für jeden der Verbraucher als gleich wahrscheinlich angesehen werden kann, dass dieser an- oder abgeschaltet war.

Will man nun das Vorhandensein eines bestimmten Verbrauchers an einem bestimmten Tag einwandfrei feststellen, kann man das Gesamtsignal auf keine vernünftige Art und Weise aufbereiten werden, da jede beliebige Kombination von ein- und ausgeschalteten Verbrauchern möglich und gleich wahrscheinlich ist. Es gibt also keine bevorzugten Fälle, die man gesondert behandeln könnte.

Deswegen kann auch mit Hilfe von Methoden aus dem Bereich des Pattern-Matching keine Lösung für das vorhandene Problem gefunden werden.

5.2.4 Fuzzy Logic und selbstlernende Algorithmen

Ein weiterer Ansatz das vorhandene Problem der Detektion eines Teilsignals aus einem Gesamtsignal zu lösen, könnte die Verwendung der so genannten Fuzzy Logic bzw. von selbstlernenden Algorithmen sein. Dadurch könnte man, so die Theorie, eine Funktion entwickeln, welche mit Hilfe von vorhandenen Datensätzen lernen könnte neuauftretende Fälle richtig einzuordnen beziehungsweise abzuschätzen. Wie man für solch eine Vorgangsweise genau vorgehen kann, kann in Fachbüchern zu diesem Thema, wie in [BAN92] und [PAL01] nachgelesen werden.

Zwei Probleme in Bezug auf die Aufgabenstellung sollen hier allerdings angesprochen werden, da sie ausschlaggebend dafür sind, warum diese Ansätze hier nicht angewandt werden. Bei beiden liegt das Problem beider für diese Algorithmen notwendigen Lernphase, denn erstens müsste diese bei der vorliegenden Aufgabenstellung für jede einzelne Filiale durchlaufen werden, wofür umfassende Daten vorhanden sein müssten. Dadurch ergäbe sich ein gewaltiger Arbeitsaufwand alleine um alle Messdaten für alle Filialen zu erhalten. Und das zweite Problem ist, dass so eine Funktion im Falle einer Änderung eines Signals, beispielsweise durch das Auswechseln eines Energieverbrauchers, die gesamte Lernphase für diese Filiale noch einmal durchlaufen müsste und danach für Daten aus der Zeit „davor“ nichtmehr zu gebrauchen wäre.

Zusätzlich stellt sich die Frage, wie sehr nicht ideale Stromverbrauchsdaten, denn darum handelt es sich im vorliegenden Fall, da ja noch keine Optimierung durchgeführt worden ist, Einfluss auf das Lernverhalten solch einer Funktion nehmen. Die Qualität der Datenbasis für solch eine Funktion ist also äußerst fragwürdig.

Aus diesem Grund kann auch aus diesen Ansätzen keine praktikable Vorgehensweise zur Lösungsfindung hergeleitet werden.

5.3 Lösung zur Erkennung einzelner Energieverbraucher

Nachdem im vorherigen Abschnitt auf alle ins Auge gefassten und wieder verworfenen Ansätze eingegangen worden ist, soll dieser Abschnitt nun eine mögliche Lösung zur Erkennung einzelner Teilsignale aus einem Gesamtsignal präsentieren.

Zuvor soll allerdings noch einmal auf die Problemstellung eingegangen werden. Es soll eine Funktion entwickelt werden, welche aus dem Tageslastprofil eines Geschäftes berechnen kann ob ein bestimmter Verbraucher an diesem Tag eingeschaltet war oder nicht. Die zugrunde liegenden Daten setzen sich dabei aus Lastprofil, Messdaten der Außentemperatur sowie Informationen über die im Geschäft vorhandenen Verbraucher zusammen. Dadurch, dass all diese Informationen vorhanden sind, stehen auch die in Kapitel 2 entwickelten Modelle für jeden bekannten Energieverbraucher zur Verfügung.

5.3.1 Erkennung des Vorhandenseins aller Verbraucher

Bevor in den nächsten Abschnitten eine mögliche Vorgehensweise zur Detektion eines einzelnen Verbrauchers vorgestellt wird, soll darauf vorbereitend, in diesem Unterkapitel eine Möglichkeit demonstriert werden, mit der abschätzbar ist ob alle Verbraucher eingeschaltet waren, oder nicht.

Dies ist deswegen interessant, da jener Fall bei dem eine gesicherte Aussage vorliegt, dass alle vorhandenen Verbraucher eingeschaltet waren, eine weitere Vorgehensweise erleichtert. Denn die Aussage „alle waren Verbraucher eingeschaltet“ inkludiert, dass jeder Verbraucher eingeschaltet war und macht eine weitere Analyse unnötig.

Zuerst soll noch einmal die Zusammensetzung des Lastprofils untersucht werden. Dieses besteht aus den einzelnen Lastprofilen einer Anzahl bekannter Energieverbraucher, welche zusammen das gesamte Lastprofil eines Geschäftes ergeben. Da die Betriebsdaten der einzelnen Verbraucher bekannt sind, können daraus die Verbrauchsmodelle der Geräte erstellt werden und aus diesen Verbrauchsmodellen kann durch simple Summenbildung auf den wahrscheinlichen Gesamtverbrauch eines Geschäftes geschlossen werden.

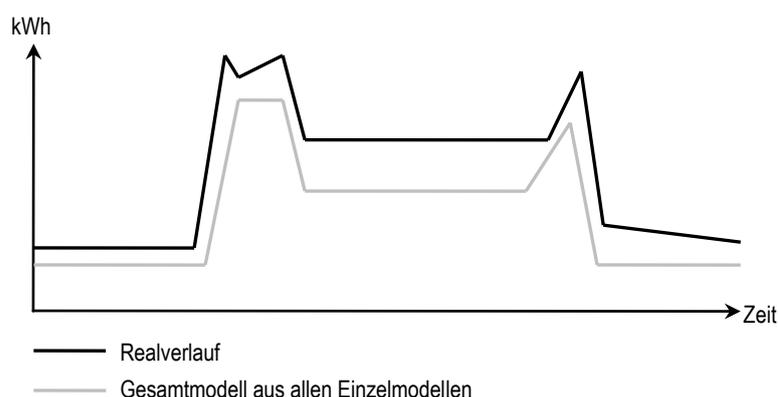


Abbildung 5.2: Gesamtmodell kleiner als Realverlauf

Addiert man also die Modelle aller Verbraucher zusammen erhält man ein Gesamtmodell für das Lastprofil eines Geschäftes. Durch Weglassen einzelner oder aller Modelle kann theoretisch jede beliebige Kombination von ein- und ausgeschalteten Verbrauchern und der sich daraus ändernde Gesamtverlauf erzeugt werden. Grundsätzlich treten all diese Kombinationen mit derselben Wahrscheinlichkeit auf, allerdings ist eine besonders hervorzuheben nämlich jene bei der alle Verbraucher eingeschaltet sind. Diese Kombination stellt den Sollzustand des Gesamtsystems dar, da in diesem Fall alle Verbraucher wie geplant eingeschaltet sind und verwendet werden. Dieser Fall, bei dem alle Verbraucher eingeschaltet sind, soll Grundlage für die weitere Überlegung sein.

Da die Modelle der Verbraucher stets das Idealmodell darstellen kann davon ausgegangen werden, dass alle Verbraucher an einem Tag gelaufen sind wenn zu allen Zeitpunkten der Modelverbrauch kleiner war als der Realverbrauch. Dieser Fall wird sehr schematisch in Abbildung 5.2 illustriert. Der Schluss, dass in diesem Fall alle Verbraucher (unter denen natürlich auch der zu detektierende ist) eingeschaltet waren kann ist deswegen zulässig, da in der Realität die Verbraucher niemals we-

niger Strom verbrauchen, als in ihrem Sollverbrauch vorgesehen. Das lässt sich deswegen so allgemein sagen, da tendenziell jeder Stromverbraucher etwas mehr Energie verbraucht, als in seinem Datenblatt angegeben.

Wird dieser Gedankengang konsequent weitergedacht kann in genau einem Fall, nämlich jenem bei dem das Gesamtmodell zu jedem Zeitpunkt des Tages kleiner als der Realverlauf ist, darauf geschlossen werden, dass alle Verbraucher eingeschaltet waren.

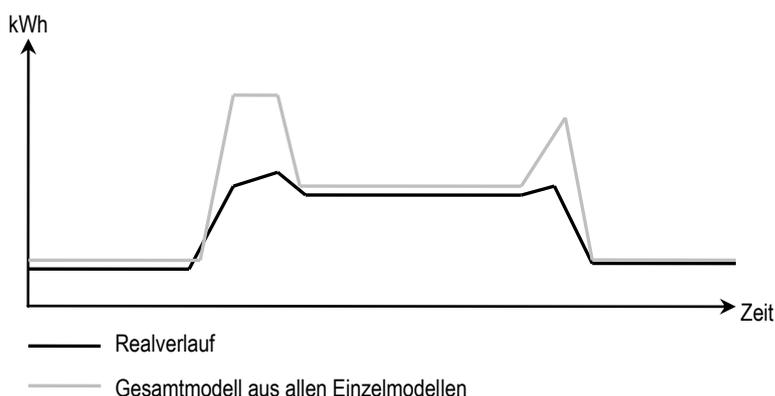


Abbildung 5.3: Gesamtmodell größer als Realverlauf

Diese Bedingung wird überprüft indem zu jedem Zeitpunkt die Differenz zwischen Real- und Modellwert gebildet wird. Ist jeder der Differenzwerte grösser als Null kann die Berechnung abgebrochen werden da alle Verbraucher eingeschaltet waren, also auch der gesuchte Verbraucher.

Ist allerdings einer der Differenzwerte negativ, dann kann daraus geschlossen werden, dass zumindest einer der Verbraucher nicht eingeschaltet war. Ein solcher Fall ist in Abbildung 5.3 dargestellt.

Nun stellt sich die Frage ob der detektierende Verbraucher mit dem nicht eingeschalteten Verbraucher übereinstimmt. Außerdem darf nicht vergessen werden, dass zu diesem Zeitpunkt nicht bekannt ist, ob einer oder mehrere Verbraucher die Abweichung verursachen. Auf diese Möglichkeiten wird im folgenden Abschnitt eingegangen.

5.3.2 Beschreibung der Vorgehensweise zur Detektion von Verbrauchern

Um nun einzelne Verbraucher detektieren zu können wird hier in einem sehr einfachen Beispiel eine mögliche Vorgangsweise vorgestellt.

In Abbildung 5.4 ist noch einmal sehr schematisch dargestellt wie sich das Gesamtmodell aus verschiedenen Verbrauchermodellen zusammensetzt. Für dieses Beispiel wurde angenommen, dass es 3 verschiedene Verbraucher gibt, welche zur selben Zeit ein- und wieder ausgeschaltet werden und jeweils einen konstanten Verbrauch haben. Gut zu erkennen ist, dass sich die Verbrauchslevels der Geräte zu einem Gesamtmodell addieren. Der Zeitpunkt zu welchem die folgende Analyse durchgeführt wird, wird dabei so gewählt, dass er in den Zeitraum zwischen Ein- und Ausschalten fällt.

Angenommen alle Verbraucher würden dem Modell entsprechend eingeschaltet sein, würde sich der Realverlauf zum Beobachtungszeitpunkt etwas über dem Gesamtmodell-Verbrauch einpendeln, da es sich bei den Modellen um Ideal-, also Minimalverbrauchsmodelle, handelt.

Was wäre aber, wenn zum Beispiel der Verbraucher 3 nicht eingeschaltet werden würde? Dann würde sich der Realverlauf ungefähr auf dem Niveau der Abgrenzung zwischen Verbraucher 2 und Verbraucher 3 einpendeln.

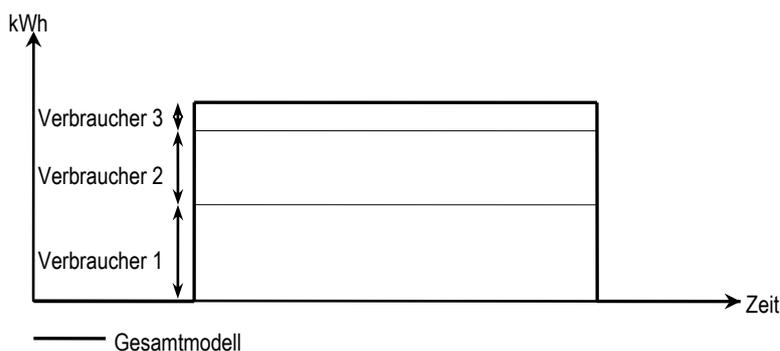


Abbildung 5.4: Schematische Zusammensetzung des Gesamtmodells

Aber auch wenn der Realverbrauch ungefähr auf diesem Level liegt und man den idealen Verbrauch aller Geräte kennt, kann daraus nicht von vornherein geschlossen werden, dass es genau Verbraucher 3 war, welcher nicht eingeschaltet worden ist. Ein Zahlenbeispiel kann das veranschaulichen:

Angenommen die Verbraucher haben folgenden Idealverbrauch:

- Verbraucher 1: 15 kWh
- Verbraucher 2: 10 kWh
- Verbraucher 3: 5 kWh

Wären all diese Verbraucher gleichzeitig eingeschaltet würde das mindesten einen Verbrauch von 30 kWh bedeuten (wie schon erwähnt ist es meist etwas mehr). Wenn allerdings z. B. nur 17 kWh gemessen werden, dann war eindeutig mindestens einer der Verbraucher nicht eingeschaltet. Nur welcher?

Um einen Verbrauch von 17 kWh zu erhalten gibt es in diesem Beispiel mehrere Kombinationen:

1. Verbraucher 1 an, Verbraucher 2 aus, Verbraucher 3 aus
2. Verbraucher 1 aus, Verbraucher 2 an, Verbraucher 3 aus
3. Verbraucher 1 aus, Verbraucher 2 aus, Verbraucher 3 an
4. Verbraucher 1 aus, Verbraucher 2 an, Verbraucher 3 an

Jeder weitere der insgesamt 8 Kombinationen (2^3) verbraucht sicher mehr als 17 kWh und ist deswegen auszuschließen.

Wäre nun gefragt ob Verbraucher 3 gelaufen ist oder nicht, könnte man folgende Aussage treffen: Von allen möglichen Fällen, die auftreten können (nämlich 4) gibt es 2 Fälle (Fall 3 und 4) die auch

die Möglichkeit inkludieren, dass Verbraucher 3 eingeschaltet war. Mit 50 prozentiger Wahrscheinlichkeit ist Verbraucher 3 also an diesem Tag eingeschaltet gewesen. Die Wahrscheinlichkeiten für Verbraucher 1 und 2 sind nach dieser Vorgehensweise 25 beziehungsweise 50 Prozent.

Es wurde also eine sehr einfache Möglichkeit gefunden, wie eine Abschätzung erhalten werden kann ob ein Verbraucher eingeschaltet war oder nicht.

5.3.3 Umsetzung für komplexe Lastprofile

Im letzten Abschnitt wurde beschrieben wie es möglich ist eine Aussage über die Wahrscheinlichkeit zu treffen, dass ein Verbraucher eingeschaltet war oder nicht. Das Beispiel, das dabei untersucht wurde, war allerdings äußerst einfach gehalten um die Kernaussage zu veranschaulichen. So wird im obigen Beispiel angenommen, dass der Beobachtungszeitpunkt innerhalb der aktiven Phase der Verbraucher liegt.

Nun zeichnen sich aber die entwickelten Modelle nicht dadurch aus, dass sie sehr einfach sind und alle zum selben Zeitpunkt die Schaltvorgänge annehmen und zwischen den beiden Zeitpunkten ein konstanter Verbrauch angenommen wird. Bei einem daraus entstehenden komplexeren Gesamtmodell stellt sich da natürlich die Frage zu welchem Zeitpunkt die Analyse durchzuführen ist. Denn es gibt zwischen Beginn und Ende der Öffnungszeiten durchaus Zeitpunkte zu denen einzelne Verbraucher bereits ausgeschaltet sind, während andere noch eingeschaltet sind.

Nun wurde in Kapitel 5.3.1 eine Möglichkeit gezeigt aus dem Lastprofil zu schließen, ob alle Verbraucher eingeschaltet waren, oder nicht. Das wurde mittels Differenzbildung zwischen Real- und Modellverlauf bewerkstelligt und aus dem Nichtvorhandensein von Differenzwerten kleiner als 0 kann man schließen, dass alle Verbraucher eingeschaltet waren. Treten nun Differenzen kleiner als 0 auf wird vermutet, dass zumindest einer der Verbraucher nicht eingeschaltet war.

Um nun einen zu analysierenden Zeitpunkt oder mehrere Zeitpunkte festzulegen werden die Differenzwerte in aufsteigender Reihenfolge sortiert. Jene Felder, bei denen die negative Differenz den größten Wert hat, werden dann der vorher vorgestellten Analyse unterzogen. Dies wird deswegen so gemacht, weil zu diesen Zeitpunkten die größte Abweichung festgestellt wurde und an diesen Zeitpunkten also wahrscheinlich die wenigsten Verbraucher eingeschaltet waren.

Hat man dann jeden dieser Zeitpunkte analysiert, können die Wahrscheinlichkeiten einfach mit Hilfe eines Mittelwertes zusammengefasst werden. So wäre es auch möglich, alle Punkte eines Tages zu analysieren und daraus einen Mittelwert über den ganzen Tag zu ermitteln, allerdings hat sich in der Praxis ein Wert von 3-5 Zeitpunkten als ausreichend genau erwiesen.

5.3.4 Mathematische Beschreibung der Vorgehensweise

Um die in den vorigen Abschnitten erläuterte Analyse automatisiert durchführen zu können bedarf es eines mathematischen Ablaufs, durch welchen es möglich ist die Vorgangsweise zu verallgemeinern.

Besonders interessant ist hier die Berechnung der Anzahl der Möglichkeiten n Modellwerte zu addieren um eine Summe kleiner oder gleich einem konstanten Wert B zu erhalten. Auf folgende Art

und Weise kann mathematisch die Aufgabe „Finde die Anzahl A aller möglichen Summen aus n Elementen, welche kleiner oder gleich einer oberen Schranke B sind.“ beschrieben werden:

$$A = \sum_{i=1}^n \lambda_B(\vec{v}_{sum,i})$$

mit

$$\lambda_B(x) = \begin{cases} 1, & x \leq B \\ 0, & x > B \end{cases}$$

und

$$\vec{v}_{sum} = \vec{v} * C$$

Die erste der oberen Formeln löst dabei die vorher gestellte Aufgabe. Die Funktion $\lambda_B(x)$ vergleicht x mit der Konstanten B und abhängig davon ob x kleiner-gleich oder größer als B ist nimmt sie den Wert 0 oder den Wert 1. Diese Funktion findet Verwendung bei einer abschließenden Summenbildung, bei der die Funktionswerte addiert werden und dadurch gezählt wird wie viele der in \vec{v}_{sum} enthaltenen Elemente einen Wert kleiner oder gleich dem konstanten Wert B haben.

\vec{v} ist dabei ein n -zeiliger Spaltenvektor, in dessen Zeilen sich die Modellwerte der verschiedenen Verbraucher, die addiert werden sollen befinden. C ist eine Matrix mit den Dimensionen $2^n \times n$, welche in jeder Zeile die Repräsentation eine mögliche Kombination von ein- und ausgeschalteten Verbrauchern enthält. Multipliziert man \vec{v} mit C erhält man wieder einen n -zeiligen Vektor \vec{v}_{sum} , dessen Elemente alle möglichen Summen der Elemente von \vec{v} enthalten.

Um nun die Wahrscheinlichkeit zu berechnen, ob ein bestimmter Verbraucher eingeschaltet war oder nicht wird die obige Formel für A zweimal angewandt. Einmal um die Anzahl der möglichen Kombinationen aller Verbraucher zu berechnen (A_{gesamt}) und ein weiteres Mal um die Anzahl aller möglichen Kombinationen von Verbrauchern zu berechnen, deren Summe kleiner als B ist und den gesuchten Verbraucher enthalten ($A_{gesucht}$). Die Wahrscheinlichkeit P , ob der Verbraucher eingeschaltet war oder nicht ergibt sich aus dem Quotienten der günstigen Fälle ($A_{gesucht}$) durch die möglichen Fälle (A_{gesamt}), also:

$$P = \frac{A_{gesucht}}{A_{gesamt}}$$

5.4 Verbesserung des Ergebnisses durch Einbeziehung der Korrelation

Mit dem in Kapitel 5.3 vorgestellten Lösungsweg sind durchaus bereits sehr gute Ergebnisse zu erzielen, was aber nicht bedeutet, dass man diese Ergebnisse nicht auch noch etwas verbessern könnte. Eine Möglichkeit das zu tun wird im folgenden Abschnitt demonstriert.

In 5.2 wurden einige mögliche Ansätze für die Detektion aufgezeigt und wieder verworfen, trotzdem kann der eine oder andere davon möglicherweise bereits berechnete Ergebnisse noch etwas verbes-

sern. Deswegen wird hier noch einmal der Korrelationskoeffizient zweier Signale genauer betrachtet.

Bei der Analyse der Daten ist aufgefallen, dass es neben den beiden am häufigsten auftretenden Fällen, nämlich dass alle Verbraucher eines Geschäftes eingeschaltet bzw. ausgeschaltet sind, noch einen gibt, der überdurchschnittlich oft vorkommt. Dabei handelt es sich um den Fall, dass alle Verbraucher bis auf einen einzelnen eingeschaltet waren.

Um nun das Ergebnis aus dem letzten Abschnitt zu verbessern, kann man folgendermaßen vorgehen: Man berechnet zuerst den Korrelationskoeffizienten zwischen einem Gesamtmodell M_1 und dem Realverlauf und danach jenen zwischen dem Realverlauf und einem Gesamtmodell M_2 , welches alle Verbraucher bis auf den zu detektierenden Verbraucher enthält.

Je nachdem welcher der beiden Korrelationskoeffizienten näher am „idealen“ Korrelationskoeffizienten 1 liegt, was bedeuten würde dass zwei Signale linear korrelieren, wird der im vorigen Abschnitt berechnete Mittelwert modifiziert. Ist der Korrelationskoeffizient zwischen Realverlauf und M_1 derjenige, der näher an 1 liegt, dann ist die Wahrscheinlichkeit grösser, dass der gesuchte Verbraucher eingeschaltet war. Ist es umgekehrt, dann ist es wahrscheinlicher, dass der gesuchte Verbraucher nicht eingeschaltet war.

Um dieses Ergebnis in den Mittelwert von vorher einbeziehen zu können, wird die Differenz zwischen den beiden Werten gebildet und dieses Ergebnis, das zwischen -1 und 1 liegen kann, wird auf einen Bereich zwischen 0 und 100 skaliert.

Dieser Wert wird dann (als weiterer „Mittelwert“) in das Ergebnis miteinbezogen. Um dem Wert etwas mehr Gewicht zu geben - wie schon erwähnt ist der Fall, dass nur ein Verbraucher nicht eingeschaltet war eindeutig der häufigste neben dem Normalfall „alle Verbraucher an“ - wird ein gewichteter Mittelwert verwendet bei dem jeder Wahrscheinlichkeitswert, welcher aus der Analyse eines Zeitpunktes entstanden ist, das Gewicht 1 und der aus den Korrelationskoeffizienten gewonnen Wert das Gewicht 2 erhält.

Dadurch kann das Ergebnis, besonders bei der Analyse weniger Punkte, entscheidend verbessert werden, während bei der Analyse mehrerer Punkte der Wert nicht so sehr in das Ergebnis einfließt, was aber auf die Genauigkeit keinen Einfluss hat, da eine Erhöhung der Genauigkeit durch die größere Menge an analysierten Zeitpunkten wettgemacht wird.

5.5 Mögliche Erweiterungen der Detektionsfunktion

Die in diesem Kapitel vorgestellte und entwickelte Möglichkeit der Detektion einzelnen Energieverbraucher aus dem Lastprofil kann natürlich nur eine erste Stufe sein, da es auch einige Kritikpunkte zu diesem Vorgehen gibt.

So erhält man zum Beispiel die besten Ergebnisse, wenn alle Verbraucher ideal oder zumindest annähernd ideal betrieben werden. Davon sollte zwar ausgegangen werden, allerdings kann es manchmal von Interesse sein bei Filialen mit einem in einem eingeschränkten Intervall sehr stark erhöhten Verbrauch (zum Beispiel in der Nacht) herauszufinden, welcher der Verbraucher wahrscheinlich

nicht deaktiviert worden ist. Für diese Erweiterung der Aufgabenstellung soll hier ein möglicher Lösungsweg skizziert werden, wobei wieder bei den Modellen der Energieverbraucher begonnen werden muss.

Zuerst muss überlegt werden ob die entwickelten Modelle der hier vorgestellten, veränderten Aufgabenstellung überhaupt gerecht werden können. Das ist gleich von Anfang an zu verneinen, da die entwickelten Modelle den Anspruch haben den idealen Stromverbrauch eines Energieverbrauchers abzubilden. Wenn man nun aber davon ausgeht, dass zum Beispiel das Licht in der Nacht nicht abgeschaltet worden ist, dann entspricht das nichtmehr den idealen Vorgaben von einem idealerweise ausgeschaltetem Licht und einem dadurch nicht vorhandenen Stromverbrauch.

Man müsste sich also zuerst überlegen wie ein Stromverbraucher sich maximal „nicht ideal“ verhalten kann, im Unterschied zum in dieser Arbeit vorgestellten Ansatz statt einem „Best-Case“-Modell ein „Worst Case“-Modell erstellen.

Diese Modellierung wäre sehr viel schwieriger vorzunehmen als die im Kapitel 2 vorgestellte, da es nur sehr wenige Möglichkeiten gibt einen Energieverbraucher ideal zu betreiben, aber sehr viele dies nicht zu tun. Bei der Beleuchtungsanlage wäre es noch relativ einfach, da der schlechteste Fall jener wäre, die Beleuchtung die ganze Nacht und den ganzen Tag aufgeschaltet zu lassen.

Bei einer Kälteanlage jedoch müsste man alle Eventualitäten in solch ein „Worst-Case“-Modell einfließen lassen und dabei hinterfragen ob solch ein Fall überhaupt realistisch ist. Denn der schlechteste Fall hier, also der maximale Stromverbrauch, würde dann auftreten, wenn bei allen Kühlregalen Tag und Nacht die Türen geöffnet wären. Dieser Fall wäre zwar eindeutig der „Worst-Case“, aber es ist wenig realistisch, dass er öfter als einmal auftritt und wenn er aufgetreten ist, ist das bekannt und die Analyse wäre obsolet.

Man sieht, um diese Modelle zu erstellen müsste sehr viel Spekulation aber auch Fachwissen in die Modellierung einfließen, denn nicht jeder Fall ist realistisch und es muss genau abgewogen werden, wie solch ein „Worst-Case“ im Einzelfall aussehen kann. Um den weiteren Lösungsweg zu skizzieren wird angenommen, dass die eben beschriebenen „Worst-Case“-Modell vorliegen.

Wenn herausgefunden werden soll, ob ein bestimmter Verbraucher die Schuld am stark erhöhten Stromverbrauch trägt, so muss die Analyse, auf dieselbe Art und Weise wie in den vorigen Abschnitten beschrieben, durchgeführt werden. Allerdings werden als Grundlage dieser Analyse nicht wie bisher die Idealmodelle, sondern die „Worst-Case“-Modelle eingesetzt.

Zu guter Letzt würde es nicht ausreichen nur einige wenige Zeitpunkte zu analysieren, schließlich kann nicht-ideales Stromverbrauchsverhalten jederzeit auftreten. Stattdessen müsste aus dem Lastprofil auf mathematische Weise berechnet werden, zu welcher Zeit die starke Abweichung zum idealen Stromverbrauch aufgetreten ist. Diese Zeitpunkte müssten dann der Analyse unterzogen werden und die berechneten Ergebnisse auf bereits beschriebene Art und Weise zu einem Mittelwert zusammengefasst werden.

Eine Verbesserung durch Einbeziehung des Korrelationskoeffizienten zwischen Modell und Realverlauf kann hier nicht durchgeführt werden, da diese Verbesserung dadurch erreicht werden konnte, dass die Daten eines ganzen Tages mit verschiedenen Gesamtmodellen verglichen wurden. Dabei wurde davon ausgegangen, dass der vorliegende Realverlauf einem idealen Verbrauchsverhalten

entspricht, allerdings ein oder mehrere Verbraucher nicht eingeschaltet waren. Wenn man sich nun die für diesen Abschnitt erweiterte Aufgabenstellung ansieht, dann wird hier von einem nicht idealen Verbrauchsverhalten innerhalb einzelner Intervalle während des Tages ausgegangen. Eine solch „umfassende“ Analyse wie jene mit dem Korrelationskoeffizienten kann also nicht zum Ziel führen.

Allerdings gäbe es noch eine andere Möglichkeit, um die hier in knapper Form umrissene Analyse anschaulicher zu machen und vielleicht bessere Ergebnisse zu erzielen:

Man belässt es nicht dabei die einzelnen Intervalle mit erhöhtem Stromverbrauch dahingehend zu überprüfen ob ein bestimmter Verbraucher für die Erhöhung verantwortlich ist, sondern führt die Analyse für jeden bekannten Verbraucher durch. Dadurch könnte man vergleichen wie wahrscheinlich es ist, dass einer der Verbraucher für den stark erhöhten Energieverbrauch verantwortlich ist und es würde eine Abschätzung ermöglicht welcher der Verbraucher nicht eingeschaltet war.

Auf diese Art und Weise könne die gefundene Lösung mit Hilfe von Wahrscheinlichkeiten aus einem Lastprofil auf das Vorhandensein oder Nichtvorhandensein einzelner Energieverbraucher zu schließen, erweitert werden.

Um die Ergebnisse dieses Kapitels zusammenzufassen, hier eine Aufzählung der vorgestellten Inhalte. Am Anfang wurde auf die Problematik der Detektion einzelnen Energieverbraucher eingegangen und danach einige der näher ins Auge gefassten Ansätze diese Aufgabe zu lösen vorgestellt. Dabei wurde auch beschrieben weshalb sie im vorliegenden Fall nicht zum Einsatz kommen. Die nächsten beiden Abschnitte stellten den entwickelten Lösungsweg vor und wie die gefundenen Lösungen mit einem bereits verworfenen Ansatz verbessert werden können. Zum Abschluss wurde noch auf eine mögliche Erweiterung eingegangen, welche auf ähnliche Art und Weise umgesetzt werden könnte.

6. Implementierung in MatLab

In diesem Kapitel wird auf einige der Besonderheiten und Probleme bei der Implementierung der für diese Diplomarbeit entwickelten Modelle und Berechnungen in MatLab/ GNU Octave eingegangen.

Alle Funktionalitäten wurden mit der freien Software GNU Octave entwickelt. Dabei handelt es sich um eine frei verfügbare Mathematik-Software, welche die für die kommerzielle Software MatLab entwickelte Befehlssprache „m“ zu 100% interpretieren kann. Dadurch gibt es zwischen für MatLab und GNU Octave erstellten Skripten keine Unterschiede. Mit Skript ist hier eine in einer Textdatei abgelegte Reihe von Befehlen gemeint, welche in beiden Programmen ausgeführt werden kann um eine einfachere Wiederholung komplexerer Befehlsfolgen zu ermöglichen.

Laut Homepage von GNU Octave (<http://www.gnu.org/software/octave/>) werden alle Unterschiede in der Interpretation von Befehlen zu MatLab, so sie bestehen, als Bugs angesehen. Um diese Aussage zu überprüfen wurde der gesamte implementierte Programmcode auch in MatLab getestet. Dabei traten keine Probleme auf. Aus diesem Grund und weil es sich bei MatLab um das „Originalprogramm“ handelt, wird in dieser Arbeit immer von „MatLab“ die Rede sein, wenn „GNU Octave bzw. MatLab“ gemeint ist.

Die Spezifikation und genaue Informationen über den erstellten Quellcode sind in Anhang A zu finden. Der gesamte erstellte Programmcode ist Anhang B zugegeben, das Urheberrecht dafür liegt bei der Envidatec GmbH.

6.1 Entwicklung spezieller Datentypen

Da die Repräsentation von Stromverbrauchern bzw. Geschäften nur sehr unzureichend durch Standarddatentypen, welche in MatLab eindeutig vorherrschend sind, erfolgen kann, war es notwendig für diesen Zweck geeignete eigene Datentypen zu definieren. Dabei musste darauf geachtet werden, dass diese Datentypen auch erweiterbar sind, falls in Zukunft andere als die momentan bekannten Energieverbraucher oder auch Geschäfte in dieses System aufgenommen und implementiert werden sollen.

In Abschnitt 6.1.1 wird gezeigt wie man auch in MatLab strukturierte Datentypen, ähnlich zu Klassen, wie sie aus der objektorientierten Programmierung bekannt sind, implementieren kann. Es wird gezeigt, dass dadurch auch die Relationen zwischen Datenobjekten abgebildet werden können, und selbst eine Veränderung der Struktur eines solchen Datensatzes sehr einfach möglich ist. Auf die

genaue Umsetzung und Verwendung dieser Technik für den erstellten MatLab-Code wird in Abschnitt 6.1.2 eingegangen.

6.1.1 Strukturierte Datentypen in MatLab

Da MatLab per se keine objektorientierte Programmiersprache ist, unterstützt sie auch keinen der Mechanismen, wie sie von objektorientierten Sprachen bekannt sind. Was aber weniger bekannt ist, ist dass es trotzdem möglich ist strukturierte Datentypen zu definieren. Solch strukturierte Datentypen stellen eine Zwischenstufe zwischen Standarddatentypen (Integer, Char, etc.) und den Objekten der höheren Sprachen dar.

In MatLab steht für solche Datentypen das so genannte *structure-Array* zur Verfügung. Es gruppiert verschiedene Datentypen, die in diesem Fall Felder genannt werden, in einem einzigen Objekt. Jedes der Felder kann dabei auch selbst ein Array mit beliebigem Inhalt sein. So kann solch ein Feld auch ein Array von *structure-Arrays* enthalten. Dadurch ist die Darstellung von *1:n* Relationen zwischen Objekten möglich. Als Beispiel könnte man den Datentyp eines Geschäftes nennen, welches mehrere Objekte eines Datentyps für Stromverbraucher enthält. Die technische Umsetzung wird bei MatLab durch die Definition eines mehrdimensionalen, assoziativen Arrays gelöst, bei dem jedes Feld (und seine Unterfelder) durch einen Schlüssel für jede der Verschachtelungsstufen ansprechbar ist.

Im Folgenden soll ein kleines Beispiel die genaue Vorgehensweise der Verwendung von strukturierten Datentypen in MatLab veranschaulichen. Es soll hier ein *structure-Array* definiert werden, welches eine Adresse repräsentiert bestehend aus zwei Strings (Name, Straßename) und einem Integer-Wert (Hausnummer). Folgendes ist dafür in die Befehlskonsole von MatLab einzugeben (oder in einer Skript-Datei abzulegen):

```
ADRESSE.name = „Klaus Pollhammer“;  
ADRESSE.strasse = „Rüdengasse“;  
ADRESSE.hausno = 3;
```

Danach ist der Datentyp definiert und es kann auf einzelne Felder mit Hilfe von „ADRESSE.feldname“ zugegriffen werden. Gibt man nur den Variablennamen der Struktur als Eingabe an wird das gesamte Objekt ausgegeben, bzw. verwendet. Wäre eines der Felder wieder ein eigener Datentyp, z. B. wenn der Datentyp aus Vor- und Nachname bestehen würde, dann könnte der Zugriff auf diese Daten so erfolgen: „ADRESSE.name.vorname“

Um die Erstellung mehrerer Variablen eines Datentyps auf möglichst effiziente Weise zu lösen, kann eine Funktion definiert werden, die diese Aufgabe erfüllt. Die Übergabeparameter der Funktion werden innerhalb dieser Funktion in die Felder des Datentyps geschrieben. Solch eine Funktion stellt in MatLab das Pendant für eine Constructor-Funktion einer Klasse dar, wie sie auch in objektorientierten Programmiersprachen (Java, C++) verwendet wird.

Um eine Funktion zu definieren muss diese einfach in einer Datei, welche denselben Namen wie die Funktion trägt, definiert und abgespeichert werden. Liegt diese Datei im Arbeitsordner von MatLab

kann sie wie eine der proprietären MatLab-Funktionen verwendet werden. In unserem Beispiel sähe die Verwendung solch einer Funktion, deren Funktionskörper in etwa den weiter oben beschriebenen Programmcode enthält mit dem Namen „makeAdress“ so aus:

```
Adresse1 = makeAdress („Klaus Pollhammer“, „Rüdengasse“, 3);
```

Trotzdem die Möglichkeit besteht etwas Ähnliches wie eine Constructor-Funktion zu definieren wird MatLab dadurch aber nicht zu einer objektorientierten Programmiersprache. So ist es nicht möglich die Eigenschaften eines auf diese Art und Weise erzeugten Objekts zu vererben, wie es in einer objektorientierten Sprache möglich wäre.

6.1.2 Datentypen für Stromverbraucher und Geschäfte

Um nun Stromverbraucher und Geschäfte in geeigneter Weise in MatLab abbilden zu können wurden die in Abbildung 6.1 gezeigten Objekte mit ihren jeweiligen Parametern implementiert. Die beiden Parameter mit den Namen „Parameter1“ und „Parameter2“ beim Objekt structDevices sind dabei die beiden Stromverbrauchslevels des Verbrauchers in der Nacht und am Tag. Bei manchen der Verbraucher kann es natürlich vorkommen, dass der Wert für den Nachtverbrauch 0 ist, da sie keinen Strom verbrauchen.

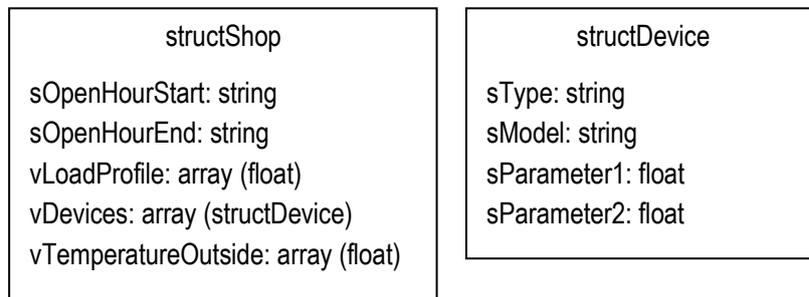


Abbildung 6.1: Die Objekte `structShop` und `structDevice`

Die Umsetzung der Datentypen wurde dem Beispiel in 6.1.1 folgend durchgeführt, wobei der Programmcode hierfür in Anhang B zu finden ist.

Es wurde bei der Implementierung darauf geachtet, dass ein Geschäft mehrere Stromverbraucher enthalten kann. Die daraus entstehende $1:n$ Beziehung wurde gewährleistet, indem eines der Felder im Datentyp `structShop` ein Array von Objekten des Typs `structDevice` ist.

6.2 Implementierung der Modelle

Um auf einfachem Weg mehrere Modelle implementieren zu können und damit die Möglichkeit zu haben bestehende Modelle zu ändern oder sogar gänzlich neue Modelle zu entwickeln wurde eine Funktion entwickelt, die aus verschiedenen Eingabeparametern das gewünschte Modell erzeugt. Die Übergabewerte der Funktion bestehen aus 5 Parametern, aus welchen die Modelldaten für jeden der bekannten Verbraucher erstellt werden können.

Bei diesen Parametern handelt es allerdings nicht um das in Kapitel 3.5 vorgestellte Parameterset. Dieses Parameterset war notwendig um die Modelle zu verallgemeinern, doch für dieses Modell-Erzeugungs-Funktion sind die Übergabewerte andere, da beispielsweise die Außentemperatur, nicht einem Verbraucher zugeordnet wurde, sondern einem ganzen Geschäft. Man könnte sagen es handelt sich um ein erweitertes Parameterset, da die einzelnen zuvor vorgestellten Parameter zwar vorhanden sind, aber nicht in der vorgestellten isolierten Form, sondern eingebettet in alle von diesem Verbraucher vorhandenen Umgebungsparameter, welche bekannt sind und zur Verfügung stehen.

Es handelt sich um folgende Parameter, eine genaue Beschreibung der Parameter erfolgt nach ihrer Aufzählung:

- Anzahl der Messpunkte pro Stunde
- Offset
- alle Informationen über das Geschäft
- Index des zu modellierenden Verbrauchers im Verbraucher-Array des Geschäftes
- Sonntags geöffnet oder nicht

Unter Offset ist hier zu verstehen um wie viele Stunden die Ergebniswerte verschoben werden sollen. Dadurch soll es möglich sein auch Messzeiträume von 02:00 bis 02:00 des folgenden Tages zu modellieren. Zu den Informationen über das Geschäft zählen alle Daten das Geschäft und die darin zu verwendenden Verbraucher betreffend (vgl. Kapitel 6.1.2). Der Index des Verbrauchers gibt an, welcher der Verbraucher modelliert werden soll. Dieser Index ist notwendig, da der Datentyp *structShop* Informationen zu sämtlichen bekannten Verbrauchern gebündelt zur Verfügung stellt und bekannt gegeben werden muss, welcher der Verbraucher nun der gewünschte ist. Außerdem muss die Funktion wissen ob es sich bei dem zu modellierenden Lastprofil um das eines normalen Arbeitstages oder das eines Sonntages handelt, da die Funktion für beide Fälle unterschiedliche Ergebnisse zurückliefern kann.

Als Rückgabewert liefert diese Verbrauchsfunktion einen Vektor, der sämtliche für den Zeitraum zwischen 00:00 und 24:00 Uhr modellierten Werte enthält (gegebenenfalls verschoben um n Stunden). Bei den modellierten Werten handelt es sich um Gleitkomma-Zahlen. Diese Werte repräsentieren die zu jedem Zeitpunkt des Tages gemessenen Realdaten, zumindest deren ideale Werte.

6.3 Implementierung der Funktion zur Berechnung einer Benchmark

Der Ansatz zur Berechnung einer Benchmark wurde schon in Kapitel 4.2 vorgestellt. Die Implementierung folgt dem dortigen Vorgehen weitestgehend. Die Funktion bekommt als Übergabeparameter die Anzahl der Messpunkte pro Stunde und eine Variable des Datentyps *structShop* (siehe 6.1.2).

Nun wird zuerst berechnet wie vielen Tagen die zu bewertenden Messdaten entsprechen. Dazu wird die Anzahl aller vorhandenen Messpunkte durch die Anzahl der Messpunkte eines Tages dividiert.

Danach wird mit Hilfe einer for-Schleife über die gesamten Messdaten iteriert, wobei bei jedem Durchlauf der Schleife, der Schleifenzähler um die Gesamtanzahl der Messpunkte eines Tages erhöht wird. Dadurch wird nicht über einzelne Messpunkte iteriert, sondern über ganze Tage.

Für jeden der Tage wird dann mit Hilfe einer Hilfsfunktion (Kapitel 6.5) ermittelt ob es sich dabei um einen Sonntag oder einen normalen Arbeitstag gehandelt hat. Das Ergebnis dieser Analyse wird in einer Boolean-Variable gespeichert.

Da die einzelnen Verbraucher unterschiedliche Modellwerte, abhängig von der bekannten Außentemperatur, haben können muss im Folgenden für jeden Tag ein Gesamtmodell aus den bekannten Verbrauchern erstellt werden.

Natürlich sind diese Modelle auch abhängig vom Ergebnis ob der Tag ein Sonntag/Feiertag war oder nicht. Dazu werden nacheinander die Werte der Verbraucher und der Temperatur (vergleiche vorherigen Abschnitt) an eine Funktion weitergereicht und die Modellwerte aller Verbraucher addiert um für jeden Tag ein Gesamtmodell zu erhalten. Der genaue Ablauf dieser Funktion ist in 6.5 beschrieben.

Liegt nun für alle Messwerte ein modellierter Wert vor, werden die Differenzwerte von Realverlauf und Modell berechnet und summiert. Die Gesamtsumme aller Differenzwerte aller Tage ergibt als Rückgabewert den Unterschied des Gesamtverbrauchs zum Idealverbrauch.

6.4 Implementierung der Funktion zur Erkennung von Energieverbrauchern

Die Funktion zur Detektion einzelner Verbraucher arbeitet weitgehend dem in Kapitel 5 entwickelten Algorithmus entsprechend. Als Übergabeparameter erhält sie einerseits eine Variable des Datentyps *structShop*, welche die gesamte Information über das Geschäft enthält, als auch eine Zeichenkette mit dem Typ des Verbrauchers um festlegen zu können welcher der Verbraucher erkannt werden soll (siehe 6.1.2).

Zu allererst wird ein Gesamtmodell für den Stromverbrauch eines Geschäftes aus den vorliegenden Daten erstellt und die Differenzwerte zwischen Realverlauf und diesem Modell berechnet. Wenn zu keinem Zeitpunkt ein Differenzwert kleiner als Null auftritt (das Gesamtmodell hat überall kleinere Werte als der Realverlauf) wird die Berechnung beendet und als Rückgabewert 100% angegeben. Gab es negative Werte werden diese in einem Array aufsteigend geordnet. Die größte Differenz befindet sich danach an erster Stelle in diesem Array, ebenso wie der zugehörige Index. Dadurch

kann herausgefunden werden zu welchem Zeitpunkt die größte negative Abweichung zwischen Idealverlauf und Realverlauf aufgetreten ist.

Nun werden die ersten n Werte in diesem Array ausgelesen nacheinander analysiert. Der Wert n ist als Konstante in der Datei festgelegt und kann frei gewählt werden. Je grösser der gewählte Wert ist, desto genauer wird die Analyse, aber auch die Belastung der Rechenkapazität steigt. Der momentane Wert liegt bei 3, was sich als ausreichend genau, bei relativ geringer Belastung der Rechenkapazität, erwiesen hat.

Um einen einzelnen Zeitpunkt zu analysieren wird zweimal im Folgenden genauer erklärte Funktion *funcNumberOfPossibleCombinations* verwendet. Diese Funktion ermittelt die Anzahl aller möglichen Kombinationen aus einer Menge von Summanden deren Summe kleiner oder gleich einem vorgegebenen Grenzwert ist.

Um auf effiziente Weise, die in Kapitel 5.3 vorgestellte mathematische Vorgangsweise zur Berechnung aller möglichen Summen, welche kleiner oder gleich einem bestimmten Wert sind, zu implementieren, wird die Spezialisierung von MatLab auf Matrizen ausgenützt. Als Übergabewerte erwartet die Funktion einen Vektor mit den Zahlen, welche zu summieren sind, und eine Zahl, bei der es sich um die Obergrenze handelt, welche nicht überschritten werden soll.

Innerhalb dieser Funktion wird zuerst ein Vektor mit so vielen Elementen erzeugt, wie es Möglichkeiten gibt die Werte zu kombinieren, nämlich 2 potenziert mit der Anzahl der Summanden. Danach wird in jedes Element des Vektors die binäre Repräsentation des Elementindex-1 geschrieben. Bei vier Zahlen und den daraus entstehenden 16 Möglichkeiten würde danach „0000“ im ersten Element und „1111“ im fünfzehnten Element stehen. Dies wird deswegen gemacht, da bei Binärzahlen alle möglichen Kombinationen von 0 und 1 verwendet werden um Zahlen gewisser Größe darzustellen und in diesem Fall alle möglichen Schaltzustände von Geräten ebenso dargestellt werden können.

Der Summanden-Vektor wird jetzt einer Vektormultiplikation mit dem gerade erzeugten Vektor aller möglichen Kombinationen unterzogen, wodurch man alle möglichen Summen der Summanden erhält.

Danach werden alle Summen, welche kleiner oder gleich Null, bzw. größer als die gesuchte Zahl sind, entfernt und die übrigen Ergebnisse gezählt. Das Ergebnis dieser Zählung wird von der Funktion als Rückgabewert geliefert.

In einem ersten Schritt wird diese Funktion verwendet um damit die Anzahl aller möglichen Kombinationen zu finden, aus dem Verbrauch aller Geräte einen Summenwert kleiner oder gleich dem Realwert zum entsprechenden Zeitpunkt zu bilden. Und ein zweites Mal um zu berechnen wie viele dieser Kombinationen den gesuchten Verbraucher beinhalten. Dafür wird der Funktion der um den Wert des gesuchten Verbrauchers verringerte Grenzwert und als mögliche Summanden alle Verbraucher außer dem gesuchten Verbraucher übergeben. Aus den beiden berechneten Werten aller Kombinationen aus allen Werten und aller Kombinationen die nur den gesuchten Verbraucher beinhalten wird die Wahrscheinlichkeit berechnet, dass der gesuchte Verbraucher zum definierten Zeitpunkt eingeschaltet war.

Wurde im Code festgelegt, dass mehr als ein Zeitpunkt analysiert werden soll, dann wird die Wahrscheinlichkeit für jeden dieser Zeitpunkte analysiert, und aus diesen Werten ein Mittelwert gebildet.

Um das Ergebnis zu verbessern werden am Ende die Korrelationskoeffizienten zwischen dem Realverlauf und dem Modell des Stromverbrauchs des gesamten Geschäftes (*KorrelationsKoeff1*) und eines leicht veränderten Gesamtmodells (*KorrelationsKoeff2*) berechnet. Bei letzterem wird angenommen, dass nur der gesuchte Verbraucher den ganzen Tag nicht eingeschaltet war, alle anderen Verbraucher jedoch ihrem Idealverlauf entsprechend verwendet wurden.

Die Differenz *KorrelationsKoeff1* minus *KorrelationsKoeff2* gibt nun Auskunft darüber ob der Korrelationskoeffizient sich verbessert hat oder nicht. Dieser Differenzwert, der zwischen -1 und 1 liegen kann, wird nun auf einen Bereich von 0-100 skaliert und ebenfalls in den Mittelwert aus den Wahrscheinlichkeiten einbezogen.

Anzumerken wäre nur noch, dass es sich bei diesem Mittelwert um einen gewichteten Mittelwert handelt, bei dem der Differenzwert der Korrelationskoeffizienten das doppelte Gewicht eines der Wahrscheinlichkeitswerte hat. Dazu werden zuerst alle Wahrscheinlichkeitswerte und die Differenz der Korrelationskoeffizienten in einem Vektor als Elemente gespeichert, die Differenz liegt dabei im letzten Element. Dann wird ein Einheitsvektor definiert, der so viele Elemente hat, wie der eben erzeugte. Der letzte Wert dieses Einheitsvektors wird auf 2 erhöht und die beiden Vektoren werden einer elementweise Multiplikation unterzogen. Dann wird die Gesamtsumme der Elemente dieses Ergebnisvektors gebildet und durch die Gesamtsumme des „Gewichtungsvektors“ dividiert, wodurch ein gewichteter Mittelwert entsteht, welcher den Rückgabewert der Funktion darstellt.

6.5 Notwendige Zusatzfunktionen

Um die in den vorigen Abschnitten vorgestellten Funktionen implementieren zu können mussten auch einige Hilfsfunktionen programmiert werden, welche für die eigentliche Funktionalität untergeordnete, aber deswegen nicht unnötige Arbeiten auslagern.

Dies hat den Vorteil, dass von verschiedenen Funktionen aus darauf zugegriffen werden kann und sie so an mehreren Stellen im Programm Verwendung finden können. Da die meisten davon sehr einfach zu verstehen sind, werden sie in diesem Text nicht behandelt, sondern sind im Anhang B zu finden. Eine davon soll allerdings schon diskutiert werden, es handelt sich dabei um eine Funktion, welche unterscheiden soll, ob ein Lastprofil von einem Sonntag (bzw. einem anderen arbeitsfreien Tag) oder einem normalen Arbeitstag mit normalen Öffnungszeiten stammt.

Funktion zur Unterscheidung von Arbeitstagen und Sonntagen

Um das Benchmarking und die Erkennung der Energieverbraucher korrekt durchführen zu können und den durch die Modellierung immer auftretenden Fehler minimieren zu können, ist eine Funktion nötig, die aus dem Stromverbrauch eines Tages erkennen kann ob es sich um einen „normalen“ Arbeitstag oder einen arbeitsfreien Sonn- bzw. Feiertag (ab hier der Einfachheit halber „Sonntag“) gehandelt hat. Das ist deswegen notwendig, da einerseits nicht von einem 7-Tage-Rhythmus mit sechs Arbeitstagen und einem freien Tag ausgegangen werden kann, schließlich können auch Feiertage während der Woche zu arbeitsfreien Tagen führen. Andererseits würde eine Nichtbeachtung der

Sonntage jede Woche zu einem nicht zu unterschätzenden Fehler führen, der nicht unbeachtet bleiben kann.

Um nun aus einem Lastprofil abschätzen zu können ob es sich beim zugehörigen Tag um einen Sonntag gehandelt hat, erstellt die Funktion aus den bekannten Daten der Verbraucher eines Geschäftes zwei Gesamtmodelle. Das eine dieser Modelle stellt den Gesamtverbrauch an einem Arbeitstag dar und das andere das Gesamtmodell an einem Sonntag. Die für das Sonntagsmodell erforderlichen Teilmodelle wurden bereits im Zuge der Modellentwicklung vorgestellt.

Nun wird zuerst der normierte mittlere Fehler zwischen dem realen Stromverbrauch des Tages und dem Modell für Sonntage und danach der normierte mittlere Fehler zwischen dem realen Stromverbrauch und dem Modell für Arbeitstage berechnet. Der normierte mittlere Fehler soll dabei, wie schon in 3.4 genauer beschrieben folgendermaßen definiert sein:

$$Fehler_{normiert} = \frac{\sum_1^n |a(n) - b(n)|}{\sum_1^n b(n)}$$

Diese Formel wurde ebenfalls in einer eigenen Funktion gekapselt, um von verschiedensten Teilen des Programms darauf zugreifen zu können. Im Fall der Sonntags-Erkennungs-Funktion wird nun zuerst der Realverlauf und das Arbeitstag-Gesamtmodell an diese Funktion übergeben und danach der Realverlauf und das Sonntags-Gesamtmodell. Die beiden Ergebniswerte werden gespeichert und verglichen.

Das Ergebnis mit dem kleineren Wert, also dem geringeren Fehler zwischen Modell und Realverlauf, bestimmt nun ob der Rückgabewert „ja“ (es war ein Sonntag) oder „nein“ (es war kein Sonntag) ist.

Hier kann es natürlich vorkommen, dass ein Tag als Sonntag erkannt wird, bei dem es sich um keinen Sonntag gehandelt hat, bzw. ein Sonntag nicht als Sonntag erkannt wird. In ersterem Fall wird ein normaler Tag ab diesem Zeitpunkt als Sonntag behandelt, im zweiten Fall ein Sonntag als normaler Tag. Im ersten Fall liegt wirklich eine Ungenauigkeit vor, allerdings wäre der Fehler in der Gesamtberechnung größer, wenn man die Sonntage nicht extra behandeln würde.

Gerade im zweiten Fall ist dieser Fehler sogar ein Vorteil für das zu analysierende Geschäft, da die in dieser Arbeit entwickelte Benchmark vor allem auf die Differenzwerte aufbaut, welche im Falle eines Sonntages, der fälschlicherweise als normaler Tag erkannt wird, sicherlich „besser“ ausfällt, als wenn das Modell eines Sonntages herangezogen würde. Dieser Fall ist nur möglich ist, wenn das Stromverbrauchsverhalten an einem solchen Tag starke Abweichungen zum Idealverbrauch zeigt und die Benchmark wäre bei einem solchen Geschäft sicherlich schlechter.

Dieser Fehler muss in Kauf genommen werden wenn man Sonntage speziell behandeln will, allerdings würde bei einer Nichtbehandlung der Sonntage der systematische Fehler eindeutig größer sein.

7. Zusammenfassung & Ausblick

Das folgende Kapitel soll eine Zusammenfassung der Diplomarbeit beinhalten und einen Ausblick darauf geben, wofür die Erkenntnisse, welche in dieser Arbeit gewonnen wurden in Zukunft verwendet werden. Zuerst wird allerdings in Kürze zusammengefasst warum die Erkenntnisse, welche im Zuge dieser Diplomarbeit gewonnen wurden, überhaupt Relevanz besitzen.

Dadurch, dass in letzter Zeit sowohl Konsumenten als auch Betriebe ein gesteigertes Problembewusstsein im Bereich der Energieproblematik und möglicher Energieeinsparungsoptionen entwickelt haben, wird es für einzelne Unternehmen immer interessanter sich auf genau diese Bedürfnisse zu spezialisieren. So suchen Unternehmen aus der Geschäftssparte der Energiedienstleister und Energy-Consulter verstärkt nach Kunden im Bereich mittelgroßer bis kleiner Betriebe, ja sogar auf Privatkunden haben sich einige dieser Betriebe spezialisiert. Das stellt einen radikalen Kurswechsel zur Situation vor weniger Jahren dar, wo es sich nur für sehr große Industriebetriebe überhaupt ausgezahlt hat Energiemanagement und Verbrauchsoptimierung zu betreiben oder auch nur Beratung auf diesem Gebiet in Anspruch zu nehmen.

Auch die Liberalisierung im Bereich der Energieversorgung trägt ihren Teil dazu bei, dass es langsam im Bewusstsein aller verankert wird, wie viel Energie und dadurch in direkter Folge Geld eingespart werden kann. Die grundlegende Vorgangsweise dafür ist meist zuerst das Stromverhaltensverhalten genau zu beobachten, das Verhalten danach zu analysieren und in Folge eine Optimierung durchzuführen. Dabei beschränken sich die potentiellen Einsparungen nicht nur auf den Bereich der elektrischen Energie, auch im Bereich der Wasserversorgung oder Beheizung kann sehr viel Geld gespart werden.

Wie wichtig das steigende Bewusstsein ist, kann sehr leicht gezeigt werden. So kann auch in Privathaushalt ein bewussteres Energieverhaltensverhalten zu ungeahnt hohen Einsparungen führen, beispielsweise wenn man die folgenden, sehr einfachen, Ratschläge befolgt:

- Niemals den Stand-By-Modus verwenden, sondern Geräte immer ganz abschalten. Vor allem Personal-Computer und Fernseher sind wahre Stromfresser.
- Alle Lampen sollten mit Energiesparlampen ausgestattet werden.
- Verwendung der „Nachtabenkung“ bei Heizungen.

Die hier beschriebenen Trends und Ratschläge können aber das Thema nicht erschöpfend behandeln, so könnte man noch etliche Beispiele anführen um zu zeigen, dass in diesen Bereichen eine wahre Aufbruchsstimmung herrscht.

Zwar nicht auf den Bereich der Privatkunden sondern auf kleinere und mittlere Betriebe hat sich die Envidatec GmbH spezialisiert. Sie bietet ihren Kunden, es handelt sich hier größtenteils um so genannte Filialkunden, neben genauer Analyse und Optimierung des Verbrauchs ein permanentes Monitoring, also eine Beobachtung oder Überwachung, einzelner Filialen an. Dabei werden die Lastprofile, wie die Tagesstromverbräuche auch genannt werden, einzelner Filialen einer Filialkette genau beobachtet und gemessen und aus diesen Beobachtungswerten wird in Folge ein genauer Optimierungsplan für die einzelnen Filialen erstellt.

Um diese Beobachtungen, also das Monitoring, effizient durchführen zu können wurde schon vor einiger Zeit von der Envidatec GmbH die Internetplattform My-JEVis entwickelt. Diese Internetplattform besteht aus einer mit Java-Applets realisierten Benutzeroberfläche und einer Datenbank. In dieser Datenbank werden mit Hilfe von Messgeräten die Lastprofile aller Standorte und Filialen der einzelnen Kunden gespeichert, welche dann über eine Benutzeroberfläche dargestellt und untereinander verglichen werden können. Auf sehr einfache Art und Weise lassen sich so die Daten der Filialen untereinander vergleichen und darstellen.

Diese Webplattform befindet sich in permanenter Weiterentwicklung und so ist als nächster Entwicklungsschritt eine Schnittstelle zwischen der Benutzeroberfläche und der Datenbank zu einem MatLab-Modul geplant. Diese Vorgehensweise soll die Auslagerung rechenintensiver Analysealgorithmen in das MatLab-Modul ermöglichen, da MatLab gerade auf Berechnungen mit großem Datenaufkommen optimiert ist. Außerdem hat eine solche Trennung den Vorteil, dass spezielle Berechnungen und Algorithmen nicht mehr nach deren eigentlicher Entwicklung in Java ausprogrammiert werden müssen, sondern in MatLab entwickelt werden können und diese Funktionen dann sehr einfach in das System integriert werden können. Inhalt dieser Arbeit war es nun für eben dieses MatLab-Modul zwei neue Analysefunktionen zu entwickeln.

Die erste der beiden Funktionen soll dabei den Energieverbrauch einzelner Filialen einer Geschäftskette benchmarken können. Dabei soll es möglich sein, eine möglichst aussagekräftige Benchmark nur aus dem Lastprofil, einigen Grunddaten und den Messdaten für die Außentemperatur zu erhalten.

Die zweite Funktion soll aus dem Lastprofil einer Filiale eines bestimmten Tages berechnen können ob ein gesuchter Verbraucher an diesem Tag eingeschaltet war oder nicht. Als Datengrundlage hier kann ebenfalls das Lastprofil, die Außentemperatur und die Daten der Energieverbraucher genannt werden.

Um die beiden Aufgaben lösen zu können wurden die Messdaten verschiedener Filialen einer Kaufhauskette aus den Jahren 2003-2007 zur Verfügung gestellt.

Im ersten Schritt war es notwendig die Daten genau zu analysieren und auch aufzuschlüsseln wie bisher bei Energieverbrauchsanalysen und -optimierungen vorgegangen wurde, worauf dabei geachtet wurde und was die Probleme und Schwierigkeiten dabei aufgetreten sind.

Dabei wurde ein besonderes Augenmerk auf die bei diesen Analysen verwendeten Modelle geworfen und gezeigt, dass diese Modelle für eine genaue Analyse nicht geeignet sind, da die einzelnen Modelle sehr ungenau sind und nicht so nahe an die Realdaten herankommen wie es wünschenswert

wäre. Es wurde auch gezeigt, dass ein nicht zu unterschätzender Fehler im Falle der Verwendung dieser Modelle auftreten kann.

Der nächste Schritt nach der Analyse der Daten war es für eine Reihe von sehr charakteristischen Energieverbrauchern allgemeine (und genauere) Modelle zu entwickeln. Dazu war es notwendig die Erkenntnisse aus der Datenanalyse so umzusetzen, dass es möglich wurde aus wenigen Parametern möglichst umfassende und genaue Modelle zu erstellen. Auf diese Weise wurden insgesamt 5 Modelle entwickelt, wobei diese Modelle durch ein einheitliches Parameterset beschreibbar sind und auch einer etwaigen Abhängigkeit des Stromverbrauchs einzelner Geräte zur Außentemperatur wurde Rechnung getragen.

Nach der Modellierungsphase wurde die erste der beiden Funktionen entwickelt. Und zwar handelte es sich dabei um eine Benchmarking-Funktion. Dazu musste allerdings herausgefunden werden wie eine Benchmark für den Energieverbrauch überhaupt realisiert werden kann und welche Informationen mit Hilfe einer solchen Benchmark vereinfacht dargestellt werden können.

Die gesuchte Funktion sollte aus dem Lastprofil eines Geschäftes eine Vergleichszahl, mit welcher Geschäfte untereinander verglichen werden können, berechnen. Dabei musste eine ganz neue Benchmark für diesen Zweck entwickelt werden, da die bisher in Verwendung befindliche nur sehr beschränkte Aussagekraft besaß. Bei dieser alten Benchmark handelte es sich um die Aufsummierung des Gesamtstromverbrauchs und eine abschließende Normierung desselben. Die Anforderungen an die neue Benchmark waren auf der einen Seite, dass es möglich war sie in variablen Abständen berechnen zu können aber auf der anderen Seite sollte trotzdem bei Addition der einzelnen Benchmarks dasselbe Ergebnis erhalten werden wie wenn nur eine Berechnung über dem „Gesamtintervall“ vorgenommen worden wäre.

Zusätzlich zu diesen Anforderungen sollte die Aussagekraft der Benchmark, also deren Qualität gesteigert werden. Die Qualitätssteigerung sollte so aussehen, dass die Funktion den realen Stromverbrauch mit dem idealen Stromverbrauch vergleichen und an Hand dieses „Wissen“ eine Bewertung durchführen sollte. Zu guter Letzt war als Anforderung gegeben, dass Benchmark unabhängig von der vollkommen unterschiedlichen Ausstattung der Filialen als Vergleichsbasis zwischen den Filialen herangezogen werden kann.

An Hand dieser Anforderungen wurde dann eine neue Benchmark entwickelt, die im Unterschied zur davor verwendeten nicht den Gesamtstromverbrauch als Vergleichszahl, sondern die Differenzwerte zwischen einem für jeden einzelnen Tag berechneten Idealverlauf und dem Realverlauf aufsummiert. Diese Summe stellt den Unterschied zwischen einem idealen Modell und dem Realverlauf dar und kann daher auch als mögliches Einsparungspotential jeder Filiale angesehen werden. Diese Vorgangsweise wird in Abbildung 7.1 schematisch dargestellt.

Auch der Fall, dass der Modellstromverbrauch größer ist als der in Wahrheit gemessene kann auf diese Art und Weise behandelt werden und hat auch eine Aussagekraft. So kann man eine negative Differenz ganz klar als Einsparung gemessen am Modellverlauf interpretieren.

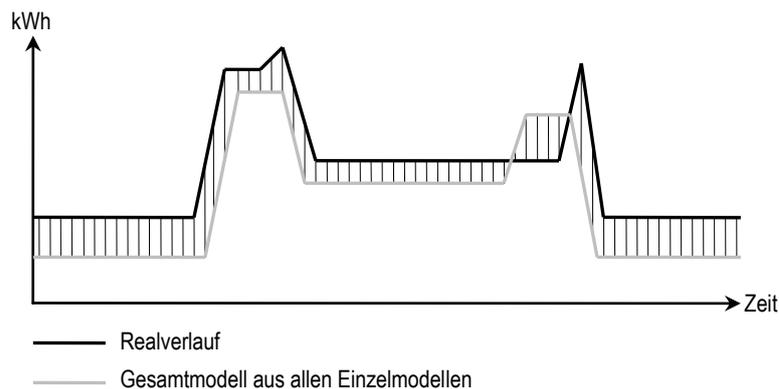


Abbildung 7.1: Differenzwerte von Realverlauf und Gesamtmodell

Das Gesamtmodell für jeden einzelnen Tag wird dabei auf Grund der bekannten Verbraucher und der Außentemperatur berechnet. Auch auf eine Erkennung von Sonn- und Feiertagen und eine entsprechende Behandlung dieser Tage wurde geachtet. Das war deswegen notwendig, da wegen des niedrigeren Stromverbrauchs an arbeitsfreien Tagen das Ergebnis der Benchmarking-Funktion stark verfälscht würde, wenn man Sonn- und Feiertage wie normale Arbeitstage behandeln würde und dadurch ein nicht unerheblicher Fehler entstehen würde.

Eine weitere Verbesserung des Ergebnisses könnte durch eine abschließende Normierung des errechneten Wertes mit dem Gesamtstromverbrauch während des zu bewertenden Zeitraums erreicht werden. Eine solche Normierung würde allerdings der Forderung widersprechen, dass durch eine Summierung der Benchmarks von aufeinanderfolgenden Zeitintervallen dasselbe Ergebnis wie bei der Bewertung des Gesamtzeitraums erreicht werden sollte.

Nach Fertigstellung der Benchmarking Funktion wurde die Entwicklung der zweiten Funktion in Angriff genommen. Mit Hilfe dieser Funktion sollte aus den Daten eines Tages-Lastprofil berechnet werden können ob ein bestimmter Verbraucher eingeschaltet war oder nicht. Zuerst wurden einige mögliche Ansätze diese Funktionalität umzusetzen untersucht und dabei herausgefunden, dass etliche Problem, welche bei Verwendung dieser Ansätze zur Lesung der vorliegenden Problemstellung auftreten können, eine ebensolche Verwendung verhindern. Dabei stellten sich einige der Ansätze als zu ungenau, andere als einfach ungeeignet heraus. Bei den untersuchten Methoden handelte es sich im Speziellen um Fourier-Analyse, Verwendung von Korrelationskoeffizienten, Pattern-Matching oder Fuzzy-Logic. All diese Ansätze wurden allerdings, wie bereits erwähnt, verworfen.

Als zielführend hat sich dann ein gänzlich neuer Ansatz herausgestellt, welcher auf der Zusammensetzung des Gesamtmodells des Stromverlaufs eines Geschäftes aus verschiedenen Einzelmodellen für die Geräte aufbaut. Bei diesem Ansatz wurde zuerst ein Gesamttagesmodell des Stromverbrauchs einer Filiale erstellt und mit dem Realverlauf verglichen. Falls das Gesamtmodell kleiner als der Realverlauf war, also keine negativen Differenzwerte auftraten, wurde darauf geschlossen, dass alle Verbraucher eingeschaltet waren, auch der gesuchte Verbraucher. Gab es allerdings einen Realverlauf, der niedriger war als der ideale Verbrauch wurde eine genauere Analyse notwendig. Diese beiden Fälle sind in Abbildung 7.2 dargestellt.

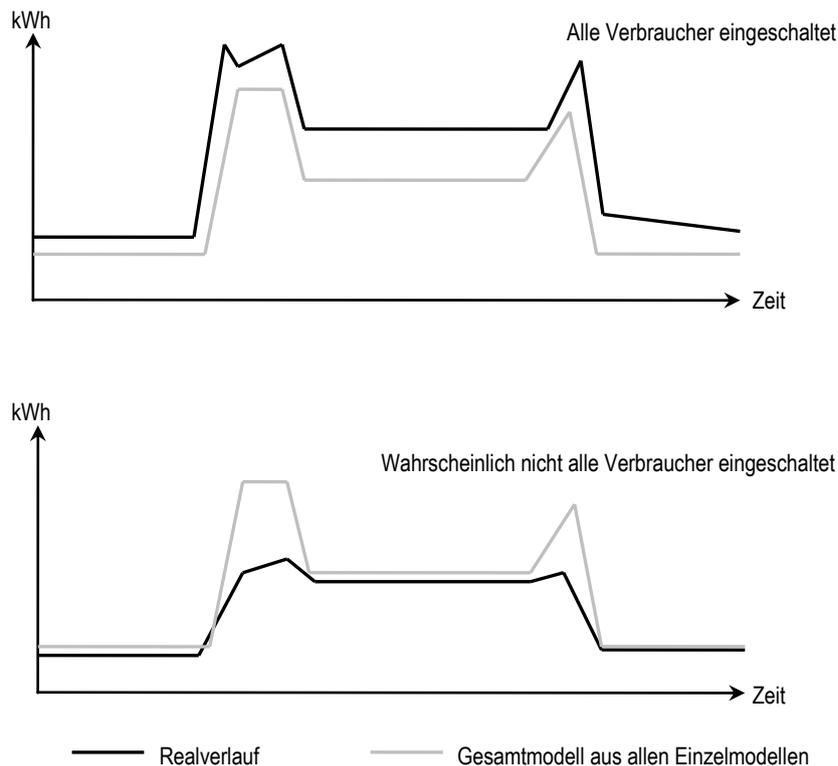


Abbildung 7.2: Entscheidungsmöglichkeiten vor der Detektionsanalyse

Ist diese erste Bewertung abgeschlossen und hat sich dabei herausgestellt, dass eine Abweichung aufgetreten ist und dadurch die Wahrscheinlichkeit besteht, dass nicht alle Verbraucher eingeschaltet waren, musste eine weitere Analyse durchgeführt werden.

Das genaue Vorgehen bei dieser weiteren Analyse kann an einem sehr einfachen Beispiel demonstriert werden um zu zeigen wie solch eine Berechnung durchgeführt werden kann. Hier in der Zusammenfassung soll dieses einfache Beispiel allerdings nicht noch einmal aufgearbeitet werden, es sei auf Kapitel 5.3.2 verwiesen. In dem einfachen, wie auch im komplizierten Fall besteht der Ansatz allerdings daraus, dass aus allen möglichen Summenkombinationen der Verbrauchswerte ermittelt wird wie hoch die Wahrscheinlichkeit ist, dass der gesuchte Verbraucher eingeschaltet war oder nicht.

Um auch komplexere Zeitverläufe analysieren zu können musste danach eine Strategie entwickelt werden, zu welchen der (bei 4 Messpunkten in der Stunde) 96 Zeitpunkten eines Tages die Berechnung überhaupt durchgeführt werden sollte. Während im einfachen Beispiel, an Hand dessen die Berechnung entwickelt wurde, eine einzige solche Analyse ausreichend war, musste für komplexere Verbrauchsverläufe eine Strategie entwickelt werden, bei welchen Zeitpunkten solch eine Analyse am zielführendsten ist. Die entwickelte Strategie fußt auf der Argumentation, dass zu jenen Zeitpunkten eines Lastverlaufs die Analyse am zielführendsten ist, an denen die größten negativen Unterschiede zwischen Modell und Lastprofil auftreten. Eine weitere Frage tauchte im Zuge dieser Überlegungen auf, nämlich die, wie viele Zeitpunkte eines Tages überhaupt überprüft werden sollten. Diese Frage ist zwar für die Güte des Ergebnisses nicht unerheblich, allerdings können auch

nicht alle Tageszeitpunkte überprüft werden, da sonst die Belastung der Berechnungskapazität zu groß wird. Dadurch, dass es allerdings gelungen ist eine Ordnung herzustellen, bei welchen Messpunkten eine Analyse am vielversprechendsten sein kann, ist es nur noch eine Frage der zur Verfügung stehenden Rechenleistung. Um die Ergebnisse aus mehreren Analysen zusammenfassen zu können wird aus den Ergebnissen mehrere Zeitpunkte wurde dann ein Mittelwert erstellt, welcher die Gesamtwahrscheinlichkeit dafür darstellte, dass der gesuchte Verbraucher gelaufen ist oder nicht.

Schon bei der Entwicklung der eben beschriebenen Strategie war die Frage aufgetaucht, ob und wie das errechnete Ergebnis noch verbessert werden könnte. Dafür wurden noch einmal die bereits verworfenen Strategien untersucht, ob die eine oder andere nicht doch hierfür verwendet werden könnte. Dabei stellte sich heraus, dass mit Hilfe der Einbeziehung des Korrelationskoeffizienten zweier Signale eine deutliche Verbesserung des Ergebnisses erzielt werden konnte. Nachdem diese neuen Erkenntnisse mit der bereits entwickelten Berechnung in Einklang gebracht werden konnten, konnte die Entwicklung der Funktion zur Erkennung einzelner Verbraucher als abgeschlossen angesehen werden und die Implementierung in Angriff genommen werden.

Diese Implementierung sollte in der befehlsorientierten Skriptsprache „m“, welche für das Programm MatLab entwickelt worden war erfolgen. Diese Skriptsprache wird allerdings nichtmehr nur vom (kommerziell vertriebenen) Programm MatLab interpretiert, sondern auch von dem, unter einer Open Source Lizenz erhältlichen, Programm „GNU Octave“. Die Implementierung der entwickelten Modelle und der beiden Funktionen erfolgte dann auch mit Hilfe dieses Programms und konnte fristgerecht Abgeschlossen werden.

Bei dieser Implementierung wurde darauf geachtet, dass alle umgesetzten Modelle (und auch die Funktionen) möglichst leicht erweitert werden können, damit auch zukünftige Entwicklungen und Verfeinerungen möglich sind, was besonders im Bereich der Modelle sehr interessant werden könnte. Dafür wurde auch eine eher selten verwendete Methode verwendet um in MatLab strukturierte Datentypen (ähnlich zu Objekten in anderen Programmiersprachen) zu ermöglichen.

Abschließend soll noch ein kurzer Blick in die Zukunft der hier vorgestellten Methoden geworfen werden. Der Schwerpunkt dabei soll allerdings nicht auf möglichen Erweiterungen beziehungsweise Verbesserungen liegen, sondern es soll eher auf den momentanen Stand der Weiterentwicklung dieser Funktionen eingegangen werden soll. Auch wie solche oder ähnliche Funktionen bei der Weiterentwicklung des Energieversorgungssektors eine größere Rolle spielen könnten soll thematisiert werden.

Wie bereits in den abschließenden Abschnitten der einzelnen Kapitel dieser Diplomarbeit besprochen gibt es für beide entwickelten Funktionen sehr unterschiedliche Möglichkeiten sie zu erweitern oder zu verbessern. So könnte man die Benchmarks durch Normierung auf die Größe des Geschäftes oder den Gesamtstromverbrauch ein bisschen besser auf die einzelnen Geschäfte anpassen, oder die Methodik zur Detektion von Energieverbraucher mit Hilfe so genannter „Worst-Case“-Modelle erweitern.

Ob solch eine Weiterentwicklung vorgenommen werden wird oder nicht, kann im Moment noch nicht abgesehen werden, da die Envidatec GmbH die Schnittstelle zwischen dem My-JEVis Web-

portal und dem MatLab-Modul eben erst fertigstellt und erste Tests mit den vorliegenden Programmteilen durchgeführt werden. Die angesprochenen Erweiterungsmöglichkeiten sind dabei die offensichtlichsten, abgesehen von der Entwicklung und Einbeziehung von neuen Modellen für die vorhandenen Energieverbraucher (z. B. das sehr vereinfachte Modell für die Lüftungsanlage) oder der Entwicklung von bisher nicht bekannten oder vernachlässigten Verbrauchern.

Falls die Entwicklung zu einem bewussteren Umgang im Energieverbrauch auf Konsumentenseite und bei den Energieversorgungsunternehmen in den nächsten Jahren nicht einen Schwenk um 180° vollführt, ist die Aussage wohl nicht allzu gewagt, dass in solchen und ähnlichen Methoden die Zukunft der Energiedienstleister und Energieerzeuger liegt. Dabei wird es sicherlich nicht bei mathematischen Abläufen zur Analyse, Detektion, Monitoring bleiben, auch die Prognose von Energieverbrauchsdaten wird in Zukunft mit rein mathematischen Algorithmen erfolgen. Dadurch wird es möglich sein, den Energieverbrauch einzelner Objekte abhängig vom momentanen Verbrauch, den momentanen Wetterdaten und den voraussichtlichen Wetterdaten aus meteorologischen Vorhersagen vorherzusagen und dadurch präventive Maßnahmen setzen zu können. Dadurch könnte man proaktiv eingreifen und wäre nicht, so wie momentan, auf eine rein reaktive Vorgehensweise angewiesen.

Inwieweit sich der Energieversorgungsmarkt in der nächsten Zeit ändern kann und wird soll nicht Teil dieser Arbeit sein, aber ebenso wie es verschiedenen Prognoseverfahren geben wird ist abzusehen, dass auch am Energieversorgungssektor immer mehr „Intelligenz“ für die Regelung der Netze und die Verteilung der Energie eingesetzt wird. Der Durchbruch der so genannten „Smart Power Grids“ kann für eine nähere Zukunft vorhergesagt werden. Solche Netze sollen so aufgebaut sein, dass eine starke Dezentralisierung der Energieversorgung ermöglicht wird und zusätzlich durch die verstärkte Verwendung von Mikroprozessoren eine höhere Flexibilität auf der Lastseite gewährleistet wird.

Wenn sich solche intelligenten Netze wirklich etablieren sollten, können Methoden, wie die in dieser Arbeit vorgestellten, dafür sorgen, dass auch auf der Verbraucherseite genügend „Intelligenz“ vorhanden ist um einen verantwortungsbewussten Umgang mit der bereitgestellten Energie zu ermöglichen.

Zwar wirkt es noch wie ein sehr großer Schritt um vom reinen Überwachen und Melden von Verbrauchsdaten zum aktiven Regeln des Energieverbrauchs eines Objektes, sei es ein Geschäft oder eine Wohnung, zu kommen. Doch bei konsequenter Weiterentwicklung der Analysewerkzeuge und Regelalgorithmen, parallel zu einer bereits vorhandenen Infrastruktur zur Datenübertragung, kann solch ein momentan noch sehr großer Schritt viel schneller getan werden als gedacht.

Specification of the MatLab/GNU Octave-Functions

Function 1 (“Detection Function”)

Function-Head

```
funcFilterDetection(p_iSPPH, p_shopS, p_sDeviceType)
```

About

The function should detect, if a specific device in a shop was turned on during a specific day. Therefore it needs information about the shop and each device that could be running in the shop.

Parameters

Parametername	Data-Type	Information
p_iSPPH	integer	Sample points per hour (p_iSPPH modulo 4 = 0 !!!)
p_shopS	structShop	Includes all information about the shop that is available.
p_sDeviceType	string	The type of the device, that should be checked if it was running or not (for example “Cooling Device”, “Ventilation” or “LightLvl1”)

Return Value

Integer value 0-100 (Percent). The probability if the device run or not.

Interval

The function should only be executed with the data of whole days, because different analyses are carried out to decide if the device was running, for example a Fourier-analysis. The data of the whole day is also needed, because the function has to decide, if the day was a "normal" day or for example a Sunday.

File

funcFilterDetection.m

Function 2 (“Benchmarking Function”)

Function-Head

```
funcFilterBenchmark(p_iSPPH, p_shopS)
```

About

The function builds a benchmark for the power-consumption of one shop, depending on information about all known devices that are used in the shop.

Parameters

Parametername	Data-Type	Information
p_iSPPH	integer	Sample points per hour (p_iSPPH modulo 4 = 0 !!!)
p_shopS	structShop	Includes all information about the shop, which is available.

Return Value

A floating-point-value that symbolizes the potential energy savings in kWh. That means that a shop is better if this value is smaller.

Interval

The function should only be executed with the data of whole days or multiples of whole days. The function benchmarks the load profile for a shop and the smallest interval for which this is possible is one day. The data of a whole day is also needed, because additionally the function has to decide, if the day was a "normal" day or for example a Sunday.

File

funcFilterDetection.m

Information about JEVIS-MatLab Functionalities

Document-Information

Version: 1.0 Author: Klaus Pollhammer
Date: 30-12-2007
Last Update: 30-12-2007

MatLab vs. GNU Octave

The functionalities were developed on GNU Octave but should be executable in MatLab without restrictions. According to the Octave-homepage are all differences between MatLab and GNU Octave considered as bugs.

About the package

The files that contain the functionalities are in the directory “matlab_files” of this package. For using them copy the files into the working-directory of MatLab/GNU Octave.

Files that start with “func” contain functions (for most function-files there is also a script-file with the prefix “test_“ for testing the function) and files that start with “struct” contain data-types. Each function-file starts with a commentary that describes what the function is for and which parameters are needed. The algorithms of the functions are described directly in the files (the algorithms for the two main-functions for detection and benchmarking are shortly summarized later in this document).

The following files are included in this package:

- Information about JEVIS-MatLab Functionalities (*this file*)
- Specification of JEVIS-MatLab Functionalities.pdf
- matlab_files
 - funcFilterBenchmark.m
 - funcFilterDetection.m
 - funcGetTimePointIndex.m
 - funcIsDaySunday.m
 - funcModelFabric.m
 - funcNormMeanError.m
 - funcNumberOfPossibleCombinations.m
 - structDevice.m
 - structShop.m
 - test_DataTypes.m
 - test_funcFilterBenchmark.m
 - test_funcFilterDetection.m
 - test_funcIsDaySunday.m
 - test_funcModelFabric.m
 - test_funcNumberOfPossibleCombinations.m

Models of shops and devices

For each known device, a model was developed. These models symbolize the optimal (minimal) power consumption that a device of a specific type has during a day. The correspondence in MatLab for these models is an array of floating-point values and is built with the function “funcModelFabric”. The models of all devices of a shop can be totalized for building a model for the power consumption of the whole shop.

Existing models

There are different types of devices and for each of these types one model was developed. If no model fits for a specific device in a shop new models can be defined easily. For new models only the files “funcModelFabric.m” and “structDevice.m” have to be adapted.

In the function “funcModelFabric” the following models are implemented:

Type	Model name	Information
Base Load	“modelBaseLoad1”	a model for the base-load of a shop
Cooling Device	“modelCool1”	a model or the cooling devices in a shop
Ventilation	“modelVentilation1”	a model for the ventilation of a shop
Light Lvl 1	“modelLightLvl1_1”	a model for the light-level 1
Light Lvl 2	“modelLightLvl2_1”	a model for the light-level 2

Data for the models

The function “funcModelFabric” has to know specific data for building a correct model for each device. For most of the models the opening hours of the shop, and two levels of the power consumption are needed (during night and day). Some need less data (for example: the light level 1 is turned off during night and has a constant consume during day), others need more (for example: the cooling device acts differently on days with high temperature).

The files “structDevice.m” and “structShop.m” contain all information about the data that is needed for the shop and each device. These two files contain also a constructor-function for the data-types.

The main functions

The description here is a short summary of what the functions do and how they do it. The commentaries directly in the files describe how the algorithms are implemented.

Specification of the functions

see: “Specification of JEVIS-MatLab Functionalities.pdf”

Detection function

This function returns how big the probability is that a specific device was running.

Attention: The result is only a probability so there is always the chance that the device was running even though the function returns a low value.

Algorithm

First the model of the power-consumption of the whole shop is built. The difference values between the real power consumption and the model are calculated. If there are no negative differences it can be assumed that all devices were running (also the one that is looked up). If there are negative values one or more devices were turned off. Now the points with the highest differences are analyzed. It is checked how many combinations of all known devices have a smaller (or equal) power consumption than the real value. After that it is calculated how many of the found combinations include the device that should be checked. Out of these two values a probability (the quotient) that represents the state of a device can be calculated and is saved for each of the analyzed points of time.

After the analysis of all points it is checked if the correlation between the real load profile and the modeled load profile gets better if the device was turned off. That would be a strong indicator if the device was turned off. Last a weighted mean value between all stored probabilities and the difference of the correlation-values is built. For this last step the correlation-difference (that lies between -1 and 1) is scaled to an interval of 0-100 because all other values are percentages.

The number of the points that should be analyzed is set with an integer-value in the function file “funcFilterDetection.m”. Directly after the start of the function the constant variable “iHowManyPointsToCheck” is set to the value 3. If the calculation stresses the performance of the computer too much also values of 1 or 2 return good results.

Benchmarking function

This function returns a benchmark for the power consumption of a shop during a period of one or more days. The smaller the value is the better was the power consumption of a shop.

Algorithm

The function checks for each day of the period if it was a Sunday or not (the models are different for Sundays/holidays). Then the difference-values between the models and the real load-profile of the shop are received. All these values are added and this sum is the benchmark that is returned. The benchmark is not only an abstract value, but has also a real pendant: the potential energy savings.

Anhang B

Im Anhang B ist der gesamte für die Envidatec GmbH erstellte Quellcode enthalten, mit Ausnahme der Files, die nur dazu erstellt worden sind um die einzelnen Funktionen zu testen.

funcFilterBenchmark.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILE:      funcFilterBenchmark.m
% VERSION:  1.0 | 2007-12-10
% AUTHOR:   Klaus Pollhammer
% ABOUT:    Calculates a benchmark for the input-data-array. For that it calculates a mod-
%           elled-time-line for the devices and compares it with the real data. The difference of the
%           real date and the model is considered as a benchmark (it really is the saving-potential of
%           the shop).
% RETURN:   Floating-Point value.
%
% PARAMETERS
% [mandatory]
%           NAME                TYPE                DESCRIPTION
%           p_iSPPH             integer         Sample Points per Hour
%           p_shopS             structShop     Includes all information that is needed about the
shop.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function fBenchmark = funcFilterBenchmark (p_iSPPH,p_shopS)

%% the benchmark is zero in the beginning
fBenchmark = 0;
model(1:24*p_iSPPH) = 0;
day = 1;

% now the benchmark is made (how
for i=1:(24*p_iSPPH):length(p_shopS.vLoadProfile)

    %% was the day a sunday?
    bSunday = funcIsDaySunday(p_iSPPH,day,p_shopS);
    model(1:24*p_iSPPH) = 0;

    %% there has to be a "temp-shop" because we only need the data of one day, so we
have to build a temporary shop
    temp_shopS = structShop (p_shopS.sOpenHourStart, p_shopS.sOpenHourEnd,
p_shopS.vLoadProfile(i:(i+24*p_iSPPH-1)), p_shopS.vDevices,
p_shopS.vTemperatureOutside(i:(i+24*p_iSPPH-1)));

    for j=1:length(p_shopS.vDevices)
        model(1:24*p_iSPPH) = model(1:24*p_iSPPH) + funcModelFabric (p_iSPPH, 0,
temp_shopS, j, bSunday);
    end

    %% now the benchmark is added to the benchmark from one step before
fBenchmark = fBenchmark + sum(p_shopS.vLoadProfile(i:(i+24*p_iSPPH-1)) - model);
day = day+1;
end
```

```

end
endfunction

```

funcFilterDetection.m

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILE:      funcFilterDetection.m
% VERSION:   1.0 | 2007-11-30
% AUTHOR:    Klaus Pollhammer
% ABOUT:     The first filter-function for the Jevis-Tool. It returns if a specific device was
%            turned on during a day.
% RETURN:    0-100 (Percent)
%
% PARAMETERS
% [mandatory]
%           NAME                TYPE                DESCRIPTION
%           p_iSPPH              integer          Sample Points per Hour
%           p_shopS              structShop     Includes all information about the shop
%           p_sDeviceType        string           The type of the device, for that should be checked
%            if it is running or not (more info see structDevice.m)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function iReturnPercent = funcFilterDetection (p_iSPPH, p_shopS, p_sDeviceType)

    %%%%%%%%%%%%%%%%%
    %
    %
    %
    % how many points do you want to check (the function will check the n smallest values)
    iHowManyPointsToCheck = 3;
    %
    %
    %
    %%%%%%%%%%%%%%%%%

    % first we reset the return value to 0 (if we cant find the device or a model that value
    will be returned)
    iReturnPercent = 0;
    iNumberOfDevices = length(p_shopS.vDevices);

    % we have to check if there is a device with the Type "p_sDeviceType" in the shop, and
    if yes, which index it has...
    iDeviceIndex = 0;
    for i=1:iNumberOfDevices
        % we look if the device with index i is the one we are searching for. if yes, we
        store the index and break the for-loop. if no we look on the next one.
        if(strcmp(p_shopS.vDevices(i).sType, p_sDeviceType))
            iDeviceIndex = i;
            break;
        end
    end

    % execute the calculations only if we found a device!
    if(iDeviceIndex > 0)
        % we have to check if the day we want to analyze was a sunday/holiday or not - the "1"
        stands, that the data of the first (and only) day should be checked (see file funcIsDaySun-
        day.m)
        bSunday = funcIsDaySunday(p_iSPPH, 1, p_shopS);

        % the next vector contains the data-points of this day (taken from the parameter)
        vLoadProfileReal = p_shopS.vLoadProfile;

        % now we will define a matrix that will contain all our models (so we only have to
        extract them once)
        vLoadProfileModels = zeros(iNumberOfDevices, 24*p_iSPPH);
        for j=1:iNumberOfDevices

```

```

        vLoadProfileModels(j,1:24*p_iSPPH) = funcModelFabric (p_iSPPH, 0, p_shopS, j,
bSunday);
    end

    % now we get the sorted differences (and there indizes) between the model and the
real data. The first value of the vectors contain the smallest difference (and so on)
    [vDifferences,vIndizes] = sort(vLoadProfileReal-sum(vLoadProfileModels));

    for(index=1:iHowManyPointsToCheck)

        if(vDifferences(index) < 0)

            fValueAtIndex = vIndizes(index);

            fRealValueAtIndex = vLoadProfileReal(fValueAtIndex);
            % all values of devices at this index are now stored into a vector
            vAllModelValuesAtIndex = vLoadProfileMo-
dels(1:iNumberOfDevices,fValueAtIndex)';

            vOtherModelValuesAtIndex = vAllModelValuesAtIndex;
            vOtherModelValuesAtIndex(iDeviceIndex)= 0;

            dividend = funcNumberOfPossibleCombinations (vOtherModelValuesAtIndex,
(fRealValueAtIndex-vAllModelValuesAtIndex(iDeviceIndex)));
            divisor = funcNumberOfPossibleCombinations (vAllModelValuesAtIndex,
fRealValueAtIndex);

            if(divisor~=0)
                vReturnPercent(index) = round(dividend/divisor*100);
            else
                vReturnPercent(index) = 0;
            end

        else
            % the models fit all into the real load profile -> our device was surely
running
            vReturnPercent(index) = 100;
        end
    end

    end

    % now we check the correlation with and without the chosen device - if it is better with-
out - that is a strong indicator, that the device was not running
    vModelWithoutDevice = vLoadProfileModels;
    vModelWithoutDevice(iDeviceIndex:iDeviceIndex,1:(24*p_iSPPH)) = [];

    % the correlation-coeff if all are running
    fCorrelCoeff1=corrcoef(vLoadProfileReal,sum(vLoadProfileModels));
    % the correlation-coeff in the case, that all except the searched device is running
    fCorrelCoeff2=corrcoef(vLoadProfileReal,sum(vModelWithoutDevice));
    % the difference between the correlation-coefficients show if it gets "better" or
"worse"
    fDiffCorrelCoeff=(fCorrelCoeff1-fCorrelCoeff2)*100;
    % the difference can be between -100 and 100 but we have to scale into an intervall from
0 to 100
    vReturnPercent(iHowManyPointsToCheck+1) = (100+fDiffCorrelCoeff)/2;

    % we will now build a weighted mean-value because the correl-coeff should be more impor-
tant than the other values (so we double it)
    vWeights(1:(iHowManyPointsToCheck+1)) = 1;
    vWeights(iHowManyPointsToCheck+1) = 2;
    vWeights = vWeights/iHowManyPointsToCheck+1;
    vReturnPercent = vReturnPercent.*vWeights;

    iReturnPercent = sum(vReturnPercent)/sum(vWeights);

endfunction

```

funcGetTimePointIndex.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILE:      funcGetTimePointIndex.m
% VERSION:  1.0 | 2007-10-30
% AUTHOR:   Klaus Pollhammer
% ABOUT:    For getting the index of the appropriate data-point for a specific time-stamp
%           The times are "translated" in the index-number of the data-points (with 4 points
per hour, 07:00 = 29 and 20:00 = 81).
%           Matlab/Octave starts to count with 1! 00:00 = 1 -> 01:00 = 5 (not 4!!!)
% RETURN:   Integer
%
% PARAMETERS
% [mandatory]
%           NAME          TYPE          DESCRIPTION
%           p_iSPPH       int           Sample Points per Hour (how many data-points are
there per hour)
%           p_sTime       string        the specific time [format: "HH:mm" - like "07:30"]
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function iTimePointIndex = funcGetTimePointIndex (p_iSPPH, p_sTime)

% the string is taken as a time-string.
vecTime = datevec(p_sTime, 'HH:MM');
% The function datevec returns an array - the field no. 4 contains the hour, the field no.
5 the minutes
vecTime = vecTime(4:5);

% the following formular returns the index we need
iTimePointIndex = vecTime(1)*p_iSPPH + (floor(vecTime(2)/(60/p_iSPPH))+1);

endfunction
```

funcIsDaySunday.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILE:      funcIsDaySunday.m
% VERSION:  1.0 | 2007-12-21
% AUTHOR:   Klaus Pollhammer
% ABOUT:    checks if a day was a sunday or not
% RETURN:   true/false
%
% PARAMETERS
% [mandatory]
%           NAME          TYPE          DESCRIPTION
%           p_iSPPH       integer       Sample Points per Hour (how many
data-points are there per hour)
%           p_iDayNumber  integer       each shop-data-type can have the
data of several days in its load-profile-vector. The integer here stands for the index of
the day...
%           p_shopS       structShop    All data about the shop that con-
sists (also all about the devices)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function bReturn = funcIsDaySunday (p_iSPPH, p_iDayNumber, p_shopS)

%% number of data-point each day?
iNumberDataPoints = 24*p_iSPPH;

%% first we get the load profile out of the shop
vRealLoadProfile = p_shopS.vLoadProfile((iNumberDataPoints*(p_iDayNumber-
1)+1):(iNumberDataPoints*(p_iDayNumber-1)+96));

%% now we get the model of all devices for sunday (and add them)
vModelSunday(1:iNumberDataPoints) = 0;
for j=1:length(p_shopS.vDevices)
    vModelSunday = vModelSunday + funcModelFabric (p_iSPPH, 0, p_shopS, j, true);
end
```

```

%% now we get the model of all devices for a normal day (and add them)
vModelNormalDay(1:iNumberDataPoints) = 0;
for j=1:length(p_shopS.vDevices)
    vModelNormalDay = vModelNormalDay + funcModelFabric (p_iSPPH, 0, p_shopS, j, false);
end

%% now we get the normalized mean error for the sunday-model (compared to the real data)
ErrSunday = funcNormMeanError(vRealLoadProfile,vModelSunday);

%% now we get the normalized mean error for the normal-model (compared to the real data)
ErrNormalDay = funcNormMeanError(vRealLoadProfile,vModelNormalDay);

bReturn = false;

%% if the error comparing to the sunday-model is smaller than the error to the normal-day-
model the probability that it was a sunday is high.
if(ErrSunday<ErrNormalDay)
    bReturn = true;
else
    bReturn = false;
end

endfunction

```

funcModelFabric.m

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILE:      funcModelFabric.m
% VERSION:   1.0 | 2007-10-30
% AUTHOR:    Klaus Pollhammer
% ABOUT:     This file contains a function, that is able to return several different model-
data-arrays of different devices. The model contains floating-point-values, that symbolize
the power consumption of the devices.
% RETURN:    Array of floating point values (length i_SPPH*24).
%
% PARAMETERS
% [mandatory]
%           NAME                TYPE                DESCRIPTION
%           p_iSPPH             integer          Sample Points per Hour (how many
data-points are there per hour)
%           p_iOffset           integer          Normally the model, that will be
returned is for a "standard" day from 00:00 to 24:00, this parameter can shift that. (1
means for 1 Hour, 2 means for 2 Hours etc.). If the Offset is 1 hour the model doesn't con-
sist of the data from 00:00 to 24:00 but from 01:00 to 01:00.
%           p_shopShop          structShop       All data about the shop that con-
sists (also all about the devices)
%           p_iDeviceIndex      integer          index of the device, that should be
modelled (information about the device lies in the vector p_shopShop.Devices())
%           p_bDayIsSunday      boolean         indicates if the model should be
for a normal day, or a "sunday" (=holiday etc.)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function vModel = funcModelFabric (p_iSPPH, p_iOffset, p_shopShop, p_iDeviceIndex,
p_bDayIsSunday)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%           The Offset is generated - the model will be shifted by this amount of datapoints.
%%%
iOffset = p_iSPPH*p_iOffset;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%           The times are "translated" in the index-number of the data-points (with 4 points per
hour, 07:00 = 29 and 20:00 = 81).
%           for more information look file:funcDetTimePointIndex.m
%%%

iOpenHourStartPoint = funcGetTimePointIndex(p_iSPPH, p_shopShop.sOpenHourStart);

```

```

iOpenHourEndPoint = funcGetTimePointIndex(p_iSPPH, p_shopShop.sOpenHourEnd);
iDataPoints = p_iSPPH*24;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   The switch-case statement is for deciding which model is wanted...
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
switch p_shopShop.vDevices(p_iDeviceIndex).sModel

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   case: Base Load Model 1
%   desc: see file:"structDevice.m"
%   additional info: The Base Load is there also on sundays
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    case 'modelBaseLoad_1'
        fAvgLoadNight = p_shopShop.vDevices(p_iDeviceIndex).fParameter1;
        fAvgLoadDay   = p_shopShop.vDevices(p_iDeviceIndex).fParameter2;

        %%% we fill the array with the data for day and night (on sundays there is no
day-base load)
            vModel(1:iDataPoints) = fAvgLoadNight;
            if(~p_bDayIsSunday)
                vModel(iOpenHourStartPoint:iOpenHourEndPoint) = fAvgLoadDay;
            end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   case: Cooling-Device Model 1
%   desc: see file:"structDevice.m"
%   additional info: Cooling devices are turned ON on sundays
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    case 'modelCool_1'

        fAvgLoadNight = p_shopShop.vDevices(p_iDeviceIndex).fParameter1;
        fAvgLoadDay   = p_shopShop.vDevices(p_iDeviceIndex).fParameter2;

        vTempOut = p_shopShop.vTemperatureOutside;

        %%% first we fill the array with the average values for day and night
            vModel(1:iDataPoints) = fAvgLoadNight;
            vModel(iOpenHourStartPoint:iOpenHourEndPoint) = fAvgLoadDay;

        %%% the model shows, that there is also power consumption before the shop opens
(2.5 hours before the day-average is inserted)
            vModel(((iOpenHourStartPoint-p_iSPPH*2.5)+1):((iOpenHourStartPoint)-1)) = fAv-
gLoadDay;

        %%% the peak in the morning is added (1.25 times the day value for 1.5 hours)
            vModel((iOpenHourStartPoint-p_iSPPH*2+1):(iOpenHourStartPoint-p_iSPPH*0.5)) =
fAvgLoadDay*1.25;

        %%% the peak in the evening is added (1.25 times the day value for 2 hours)
            vModel((iOpenHourEndPoint+1-p_iSPPH*1.5):(iOpenHourEndPoint+p_iSPPH*0.5)) =
fAvgLoadDay*1.25;

        %%% the temperature-offset (only if day-maximum-temperature is higher than 15 °C
            TempFactor = max(vTempOut)-15;
            if(TempFactor > 0)
                vModel = vModel.+(TempFactor*0.5);
            end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   case: Ventilation Model 1
%   desc: see file:"structDevice.m"
%   additional info: Ventilation is turned OFF on sundays
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    case 'modelVentilation_1'

        fLevel1 = p_shopShop.vDevices(p_iDeviceIndex).fParameter1;
        %%% first we fill the array with the average values for day and night

```

```

                vModel(1:iDataPoints) = 0; %the ventilation
should not consume any power during the night

        %% the ventilation consumes only power, if the day is a "normal" day and not a
sunday (or holiday)!
        if(~p_bDayIsSunday)
        %% the ventilation is turned on between the start and the end of the opening
hours
                vModel(iOpenHourStartPoint:iOpenHourEndPoint) = fLevel1; % the "normal" Level
during day

        %% model includes, that one hour before shop opens the ventilation is started
vModel((iOpenHourStartPoint-p_iSPPH*1)+1):(iOpenHourStartPoint-1)) = fLe-
vell;

        end

        %%%%%%%%%%%
        % case: Lights Level 1 Model 1
        % desc: see file:"structDevice.m"
        % additional info: Lights are turned OFF on sudays
        %%%

        case 'modelLightLvl1_1'

                fLevel1 = p_shopShop.vDevices(p_iDeviceIndex).fParameter1;

                %% first we fill the array with the values for the night (lights are turned off
during night so power consumption is zero!)
                vModel(1:iDataPoints) = 0;

                %% the light consumes only power, if the day is a "normal" day and not a sunday
(or holiday)!
                if(~p_bDayIsSunday)
                %% now level 1 is turned on 1.5 hours before shop opens and 1.5 hour after it is
closed its turned off
                        vModel((iOpenHourStartPoint-p_iSPPH*1.5)+1):(iOpenHourEndPoint +
p_iSPPH*1.5)) = vModel((iOpenHourStartPoint-p_iSPPH*1.5)+1):(iOpenHourEndPoint +
p_iSPPH*1.5)) + fLevel1;
                end

                %%%%%%%%%%%
                % case: Lights Level 2 Model 1
                % desc: see file:"structDevice.m"
                % additional info: Lights are turned OFF on sudays
                %%%

                case 'modelLightLvl2_1'

                        fLevel2 = p_shopShop.vDevices(p_iDeviceIndex).fParameter1;

                        %% first we fill the array with the values for the night (lights are turned off
during night so power consumption is zero!)
                        vModel(1:iDataPoints) = 0;

                        %% the light consumes only power, if the day is a "normal" day and not a sunday
(or holiday)!
                        if(~p_bDayIsSunday)
                        %% level 2 should be turned on between 15min before and after the opening hours
                                vModel((iOpenHourStartPoint-p_iSPPH*0.25)+1):(iOpenHourEndPoint +
p_iSPPH*0.25)) = vModel((iOpenHourStartPoint-p_iSPPH*0.25)+1):(iOpenHourEndPoint +
p_iSPPH*0.25)) + fLevel2;
                        end

                        %%%%%%%%%%%
                        % case: Unknown Device
                        %%%

                otherwise

```

```

        end % end switch p_shopShop.vDevices(p_iDeviceIndex).sModel

        vModel = vModel([ end-iOffset+1:end 1:end-iOffset ]);
endfunction

```

funcNormMeanError.m

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILE:      funcNormMeanError.m
% VERSION:  1.0 | 2007-12-04
% AUTHOR:   Klaus Pollhammer
% ABOUT:    Calculates the normalized mean error between two vectors
% RETURN:   Floating-point value between 0 and 1
%
% PARAMETERS
% [mandatory]
%          NAME                TYPE
%          p_vA                vector
%          p_vB                vector
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function R = funcNormMeanError (p_vA,p_vB)
%% we take the vector of the absolut differences and sum them all up
dividend = sum(abs(vA-vB));
%% now we take all values of vector b, square them and sum them up
divisor = sum(vB);
%% our return value is the square-root of the quotient
R = sqrt (dividend/divisor);
endfunction

```

funcNumberOfPossibleCombinations.m

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILE:      funcNumberOfPossibleCombinations.m
% VERSION:  1.0 | 2007-12-18
% AUTHOR:   Klaus Pollhammer
% ABOUT:    Returns the number of all possibilities to add the elements of an array with a
lower sum than a given value
% RETURN:   Integer
% PARAMETERS
% [mandatory]
%          NAME                TYPE                DESCRIPTION
%          p_vValues          vector            This vector includes the elements for the summa-
tion.
%          p_fMax             float            This is the maximum-value, that should be checked.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function iReturnNumber = funcNumberOfPossibleCombinations (p_vValues, p_fMax)

    m = dec2bin((0:((2^length(p_vValues))-1)));
    m_size = size(m);

    % now we make a 1/0 array with all possible combinations (the minus 48 is because the val-
ues are converted in the ascii-values and we have to decrease the numbers by 48 to get 1 and
0)
    vAllPossibleCombinations=abs(m(1:m_size(1),1:m_size(2))-48);

    % now we multiply our values with the combinations and get all sums
    vAllSums=p_vValues*vAllPossibleCombinations';

    % we get rid of all possibilities, that only contain zero
    vAllSums = nonzeros(vAllSums');

    % now we compare all values with our maximum - and get rid of all zeros (only the values
smaller or equal are interesting for us)
    vSumsSmallerThanValue = nonzeros(vAllSums<=p_fMax);

```

```

    iReturnNumber = length(vSumsSmallerThanValue);
endfunction

```

structDevice.m

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILE:      structDevice.m
% VERSION:  1.2 | 2007-12-06
% AUTHOR:   Klaus Pollhammer
% ABOUT:    This file contains information about the data-structure, that represents a de-
%           vice, and how the functions will assume, that the data looks like!
%           The file also contains a function that returns the correct data-type.

% DEFINITION OF THE DATATYPE
%
% structDevice.sType      = 'string'      String, that holds the type of the device. (de-
%                               vices of the same type could use different models)
% structDevice.sModel    = 'string'      String, that holds the name of the model that
%                               you want to use.
% structDevice.fParameter1 = 'float'      Floating-point value, that holds (for Example)
%                               the average power consumption during the night [kWh]!
% structDevice.fParameter2 = 'float'      Floating-point value, that holds (for Example)
%                               the average power consumption during the day [kWh]!
%
% EXAMPLES
%
% The structDevice-Datatype is used for the following devices:
%
% Type: Base Load Model 1
%   structDevice.sType      = 'Base Load'
%   structDevice.sModel    = 'modelBaseLoad1'
%   structDevice.fParameter1 = Base Load during the night
%   structDevice.fParameter2 = Base Load during the day
%
% Type: Cooling Device Model 1
%   structDevice.sType      = 'Cooling Device'
%   structDevice.sModel    = 'modelCool1'
%   structDevice.fParameter1 = Average power consumption during the night
%   structDevice.fParameter2 = Average power consumption during the day
%
% Type: Ventilation Model 1
%   structDevice.sType      = 'Ventilation'
%   structDevice.sModel    = 'modelVentilation1'
%   structDevice.fParameter1 = Power consumption of ventilation-level1
%   structDevice.fParameter2 = Power consumption of ventilation-level2
%
% Type: Lights Level 1 Model 1
%   structDevice.sType      = 'LightLvl1'
%   structDevice.sModel    = 'modelLightLvl1_1'
%   structDevice.fParameter1 = Average power consumption first light-level (1/3-light)
%   structDevice.fParameter2 = not used
%
% Type: Lights Level 2 Model 1
%   structDevice.sType      = 'LightLvl2'
%   structDevice.sModel    = 'modelLightLvl2_1'
%   structDevice.fParameter1 = Average power consumption second light-level (2/3-light)
%   structDevice.fParameter2 = not used
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% FUNCTION: the function takes all the parameters, and makes a "correct" structure out of
% them.
% PARAMETERS
% [mandatory]
%
%           NAME                TYPE                DESCRIPTION
%           p_sType              string            Defines the overall device-type (devices of the
% same type could use different models)

```

```

%           p_sModel           string      Name of the model that should be used (for exam-
ple: 'light1')
%           p_fParameter1      float      Parameter 1 for the model (information about that,
see above)
% [optional]
%           p_fParameter2      float      Parameter 2 for the model (information about that,
see above - ATTENTION: some models need that value!!)
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function D = structDevice (p_sType, p_sModel, p_fParameter1, p_fParameter2)

    if nargin==3, p_fParameter2 = 0; end

    D.sType       = p_sType;
    D.sModel      = p_sModel;
    D.fParameter1 = p_fParameter1;
    D.fParameter2 = p_fParameter2;

endfunction

```

structShop.m

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FILE:      structShop.m
% VERSION:   1.0 | 2007-11-28
% AUTHOR:    Klaus Pollhammer
% ABOUT:     This file contains information about the data-structure, that represents a shop,
and how the functions will assume, that the data looks like!
%           The file also contains a function that returns the correct data-type.
%
% DEFINITION OF THE DATATYPE
%
% structShop.sOpenHourStart      = 'string'      When does the shop open? Format:
HH:MM - like '08:00'
% structShop.sOpenHourEnd        = 'string'      When does the shop close? Format:
HH:MM - like '20:00'
% structShop.vLoadProfile         = 'vector'      15min load course of the shop
(floating point values) [kWh]!
% structShop.vDevices             = 'vector'      Information about the devices, that
are known to be in the shop will be in here. (for information about the data-type see file
'structDevice').
% structShop.vTemperatureOutside  = 'vector'      The outside-temperature 15min data
samples (floating-point values)
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FUNCTION:  the function takes all the parameters, and makes a "correct" structure out of
them.
% PARAMETERS
% [mandatory]
%           NAME                TYPE           DESCRIPTION
%           p_sOpenHourStart     string      see above
%           p_sOpenHourEnd       string      see above
%           p_vLoadProfile        vector      A vector consisting of structDevice-
Elements.
%           p_vDevices            vector      see above
%           p_vTemperatureOutside vector      see above
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function S = structShop (p_sOpenHourStart, p_sOpenHourEnd, p_vLoadProfile, p_vDevices,
p_vTemperatureOutside)

    S.sOpenHourStart = p_sOpenHourStart;
    S.sOpenHourEnd   = p_sOpenHourEnd;
    S.vLoadProfile    = p_vLoadProfile;
    S.vDevices        = p_vDevices;
    S.vTemperatureOutside = p_vTemperatureOutside;

endfunction

```

Anhang C

In Kapitel 3.4 wurde der normierte mittlere Fehler zwischen neuem bzw. altem Modell und den Realwerten von zwei Filialen zu jeweils 3 Tagen berechnet. Die folgenden Tabellen beinhalten alle für diese Berechnungen notwendigen Werte. Bei den als „real“ beschrifteten Spalten handelt es sich um die Messdaten eines Stromzählers an einer Kälteanlage. Es gibt immer nur eine Spalte für das alte Modell, da dieses früher unabhängig von der Außentemperatur war und deswegen keinen Schwankungen wie das neue Modell unterworfen war. Die Spalte „Temp“ bezeichnet die Messwerte für die Außentemperatur.

Werte Filiale 1

Filiale 1										
Zeit	16.1.			10.8.			22.12.			Modell alt kWh
	real kWh	Temp °C	Modell neu kWh	real kWh	Temp °C	Modell neu kWh	real kWh	Temp °C	Modell neu kWh	
00:00	20	3	18,75	25,4	13,2	19,95	20,2	-0,1	18,75	18,75
00:15	20	3,2	18,75	25,6	13,4	19,95	20,4	-0,4	18,75	18,75
00:30	20	3,2	18,75	27	13,2	19,95	20,2	-0,4	18,75	18,75
00:45	21	3,2	18,75	30,8	13,1	19,95	21,2	-0,4	18,75	18,75
01:00	22	2,9	18,75	29,4	13,1	19,95	22	-0,4	18,75	18,75
01:15	22	2,9	18,75	29,2	12,9	19,95	22	-0,4	18,75	18,75
01:30	22	3	18,75	28,4	12,7	19,95	21,8	-0,4	18,75	18,75
01:45	21	2,9	18,75	27,4	12,6	19,95	21,8	-0,6	18,75	18,75
02:00	22	2,9	18,75	28,2	12,6	19,95	21,8	-0,8	18,75	18,75
02:15	21	2,9	18,75	26,4	12,6	19,95	22	-0,6	18,75	18,75
02:30	21	2,9	18,75	28,2	12,4	19,95	22	-0,4	18,75	18,75
02:45	22	2,7	18,75	27,2	12,3	19,95	22,6	-0,4	18,75	18,75
03:00	22	2,7	18,75	29	12,1	19,95	22,8	-0,4	18,75	18,75
03:15	22	2,7	18,75	28,2	12,1	19,95	21,8	-0,6	18,75	18,75
03:30	22	2,7	18,75	25,6	12	19,95	21,8	-0,6	18,75	18,75
03:45	21	2,7	18,75	27,8	11,8	19,95	22,2	-0,6	18,75	18,75
04:00	22	2,7	18,75	27,4	11,6	19,95	23	-0,6	18,75	18,75
04:15	25	2,7	18,75	35,4	11,5	19,95	22,6	-0,6	18,75	18,75

Filiale 1										
Zeit	16.1.			10.8.			22.12.			Modell alt kWh
	real kWh	Temp °C	Modell neu kWh	real kWh	Temp °C	Modell neu kWh	real kWh	Temp °C	Modell neu kWh	
04:30	32	2,7	18,75	48,6	11,3	19,95	30,4	-0,6	18,75	18,75
04:45	42	2,7	31,25	46,8	11,3	32,45	42	-0,6	31,25	18,75
05:00	42	2,7	31,25	54,8	11,3	32,45	41,6	-0,6	31,25	18,75
05:15	49	2,7	39,06	44	11,3	40,263	51,2	-0,4	39,063	18,75
05:30	35	2,7	39,06	41,6	11,2	40,263	35,8	-0,4	39,063	18,75
05:45	37	2,7	39,06	38,2	11,2	40,263	38,6	-0,3	39,063	18,75
06:00	32	2,7	39,06	38,4	11,2	40,263	34,6	-0,3	39,063	18,75
06:15	34	2,5	39,06	38,6	11,2	40,263	30	-0,3	39,063	18,75
06:30	31	2,5	39,06	38	11,2	40,263	30,6	-0,3	39,063	31,25
06:45	31	2,4	31,25	44,8	11,2	32,45	31,8	-0,3	31,25	31,25
07:00	34	2,4	31,25	42	11,2	32,45	33,8	-0,1	31,25	31,25
07:15	34	2,4	31,25	41,2	11,2	32,45	33	0	31,25	31,25
07:30	34	2,2	31,25	41	11,5	32,45	32,6	0	31,25	31,25
07:45	33	2,2	31,25	41	11,8	32,45	33,4	0	31,25	31,25
08:00	31	2,2	31,25	40,4	12	32,45	32,8	0	31,25	31,25
08:15	31	2,2	31,25	40,4	12,4	32,45	31,8	0	31,25	31,25
08:30	32	2,2	31,25	38	12,4	32,45	32	0,2	31,25	31,25
08:45	29	2,2	31,25	40,4	12,7	32,45	29	0,2	31,25	31,25
09:00	28	2,2	31,25	39,2	13,1	32,45	28,8	0,2	31,25	31,25
09:15	30	2,2	31,25	39,4	13,2	32,45	28,8	0,5	31,25	31,25
09:30	29	2,2	31,25	41,2	13,7	32,45	31,6	0,5	31,25	31,25
09:45	29	2,4	31,25	39	14	32,45	32,2	0,5	31,25	31,25
10:00	29	2,4	31,25	41,8	14,5	32,45	31,2	0,7	31,25	31,25
10:15	31	2,4	31,25	42,2	14,6	32,45	32,4	0,7	31,25	31,25
10:30	33	2,5	31,25	41,6	15,1	32,45	30,2	0,8	31,25	31,25
10:45	30	2,5	31,25	42,6	15,1	32,45	30	1	31,25	31,25
11:00	33	2,5	31,25	42,2	15,2	32,45	30,2	1	31,25	31,25
11:15	30	2,5	31,25	42,6	15,6	32,45	32	1	31,25	31,25
11:30	31	2,7	31,25	43,2	15,9	32,45	33	1	31,25	31,25
11:45	31	2,7	31,25	43,2	16	32,45	29,4	1,1	31,25	31,25
12:00	29	2,9	31,25	42,6	16	32,45	32,2	1,1	31,25	31,25
12:15	31	2,9	31,25	43,8	16,3	32,45	31,8	1,1	31,25	31,25
12:30	32	2,9	31,25	43,6	16,5	32,45	29,6	1,1	31,25	31,25
12:45	28	2,9	31,25	43,2	16,7	32,45	30	1,1	31,25	31,25
13:00	29	3	31,25	44,8	16,7	32,45	29	1,1	31,25	31,25
13:15	32	3	31,25	44,2	16,7	32,45	29,2	1,1	31,25	31,25
13:30	31	3,2	31,25	42,4	16,8	32,45	30,4	1,3	31,25	31,25
13:45	30	3,2	31,25	43,2	17	32,45	31,8	1,3	31,25	31,25

Filiale 1										
Zeit	16.1.			10.8.			22.12.			Modell alt kWh
	real kWh	Temp °C	Modell neu kWh	real kWh	Temp °C	Modell neu kWh	real kWh	Temp °C	Modell neu kWh	
14:00	33	3	31,25	43,2	17	32,45	32,2	1,4	31,25	31,25
14:15	30	3	31,25	42,8	17	32,45	32,2	1,4	31,25	31,25
14:30	29	3	31,25	43,8	17	32,45	31,2	1,4	31,25	31,25
14:45	30	3	31,25	45	17,3	32,45	31,6	1,4	31,25	31,25
15:00	32	2,9	31,25	43	17,1	32,45	33,6	1,4	31,25	31,25
15:15	31	2,9	31,25	43,8	17,3	32,45	30,4	1,3	31,25	31,25
15:30	29	2,9	31,25	43,4	17,4	32,45	29	1,4	31,25	31,25
15:45	32	2,9	31,25	43,2	17,3	32,45	29,6	1,4	31,25	31,25
16:00	29	2,9	31,25	43,2	17,4	32,45	31,8	1,4	31,25	31,25
16:15	30	2,9	31,25	43	17,3	32,45	32,2	1,6	31,25	31,25
16:30	32	2,7	31,25	42,8	17,1	32,45	31,8	1,6	31,25	31,25
16:45	31	2,4	31,25	43	17	32,45	32,2	1,6	31,25	31,25
17:00	29	2,2	31,25	43,2	17	32,45	30,8	1,8	31,25	31,25
17:15	32	2,4	31,25	42,4	16,8	32,45	29	1,8	31,25	31,25
17:30	30	2,4	31,25	42,6	16,8	32,45	28,6	1,9	31,25	31,25
17:45	29	2,2	31,25	39,6	16,5	32,45	31	2,1	31,25	31,25
18:00	31	2,2	31,25	40,4	16,5	32,45	33	2,1	31,25	31,25
18:15	29	2,1	31,25	43,2	16,5	32,45	31	2,1	31,25	31,25
18:30	29	1,9	31,25	42,6	16,5	32,45	31,2	2,1	31,25	31,25
18:45	28	1,9	39,06	41,6	16,3	40,263	29,4	2,1	39,063	31,25
19:00	31	1,8	39,06	43,2	16,2	40,263	28,4	2,2	39,063	31,25
19:15	33	1,8	39,06	44,6	16	40,263	33,4	2,2	39,063	31,25
19:30	31	1,8	39,06	42,8	15,9	40,263	29,6	2,4	39,063	31,25
19:45	28	1,6	39,06	41,4	15,7	40,263	29,2	2,4	39,063	31,25
20:00	33	1,4	39,06	49	15,7	40,263	31	2,4	39,063	31,25
20:15	44	1,4	39,06	48,8	15,4	40,263	43,8	2,4	39,063	18,75
20:30	39	1,3	39,06	48,2	15,4	40,263	40,6	2,5	39,063	18,75
20:45	45	1,4	18,75	49,4	15,2	19,95	43,8	2,7	18,75	18,75
21:00	38	1,4	18,75	54	15,2	19,95	39,6	2,7	18,75	18,75
21:15	38	1,4	18,75	47,8	15,2	19,95	42	2,9	18,75	18,75
21:30	35	1,4	18,75	50,6	15,1	19,95	34	2,9	18,75	18,75
21:45	39	1,4	18,75	47,2	14,9	19,95	41,8	2,9	18,75	18,75
22:00	34	1,3	18,75	37,8	14,8	19,95	36,2	2,9	18,75	18,75
22:15	28	1,3	18,75	32,6	14,6	19,95	28	2,9	18,75	18,75
22:30	25	1,3	18,75	31,6	14,5	19,95	25	2,9	18,75	18,75
22:45	26	1,3	18,75	33,2	14,5	19,95	26,2	3	18,75	18,75
23:00	26	1,1	18,75	35	14,3	19,95	25,4	3	18,75	18,75
23:15	25	1,1	18,75	32,4	14,2	19,95	25,8	3,2	18,75	18,75

Filiale 1										
Zeit	16.1.			10.8.			22.12.			Modell alt kWh
	real kWh	Temp °C	Modell neu kWh	real kWh	Temp °C	Modell neu kWh	real kWh	Temp °C	Modell neu kWh	
23:30	27	1,1	18,75	29,4	14	19,95	24,6	3,2	18,75	18,75
23:45	22	1,1	18,75	25,6	13,8	19,95	24	3,2	18,75	18,75

Werte Filiale 2

Filiale 2										
Zeit	24.1.			16.7.			22.10.			Modell alt kWh
	real kWh	Temp °C	Modell neu kWh	real kWh	Temp °C	Modell neu kWh	real kWh	Temp °C	Modell neu kWh	
00:00	16,5	-7,4	10	21	24,6	21,4	18,25	2,2	10	10
00:15	17,25	-7,4	10	20,25	24,3	21,4	18,25	2	10	10
00:30	16,5	-7,3	10	19,5	24,1	21,4	17,5	1,8	10	10
00:45	16	-7,1	10	19,5	23,9	21,4	18,5	1,7	10	10
01:00	15,5	-7	10	19,5	23,7	21,4	17,25	1,6	10	10
01:15	16,25	-7,3	10	19,25	23,7	21,4	17,5	1,6	10	10
01:30	17,5	-7,4	10	22	23,4	21,4	18,25	1,6	10	10
01:45	17,75	-7,6	10	21,75	23,3	21,4	21,75	1,4	10	10
02:00	17,75	-7,7	10	21,75	23,1	21,4	17,75	1,2	10	10
02:15	15,25	-7,7	10	21,5	22,9	21,4	20	1,1	10	10
02:30	17,5	-7,3	10	20,25	22,7	21,4	18,25	0,9	10	10
02:45	15	-6,9	10	19,75	22,4	21,4	18,25	0,9	10	10
03:00	16,25	-6,4	10	18,75	22,3	21,4	17,25	0,8	10	10
03:15	16	-6,3	10	19,75	22,2	21,4	17,75	0,6	10	10
03:30	15,5	-6,9	10	18,75	22	21,4	17,5	0,4	10	10
03:45	16	-7,4	10	18	21,8	21,4	16,75	0,2	10	10
04:00	16	-7,7	10	20	21,6	21,4	16,5	0,2	10	10
04:15	15,75	-7,7	10	20	21,3	21,4	18,5	0,3	10	10
04:30	18,5	-7,7	10	20	21,2	21,4	17,5	0,5	10	10
04:45	18,5	-7,7	16	21,25	21,1	27,4	20	0,6	16	10
05:00	14	-7,7	16	19,75	20,9	27,4	19,75	0,6	16	10
05:15	18,25	-7,7	20	19,5	21,2	31,4	17,75	0,5	20	10
05:30	20,5	-7,8	20	22,75	21	31,4	19	0,3	20	10
05:45	24,25	-7,9	20	27,75	20,8	31,4	21,5	0,2	20	10
06:00	25,75	-7,9	20	29,75	20,7	31,4	27	0,1	20	10
06:15	19,5	-7,9	20	25,25	20,6	31,4	26	0	20	10
06:30	25,75	-7,9	20	25,75	20,6	31,4	21,75	-0,1	20	16

Filiale 2										
Zeit	24.1.			16.7.			22.10.			Modell alt kWh
	real kWh	Temp °C	Modell neu kWh	real kWh	Temp °C	Modell neu kWh	real kWh	Temp °C	Modell neu kWh	
06:45	25,5	-7,9	16	30,25	20,8	27,4	27	-0,2	16	16
07:00	19,5	-8	16	29	21,6	27,4	27,25	-0,3	16	16
07:15	19,75	-8	16	25,5	21,9	27,4	21,75	-0,3	16	16
07:30	23,25	-8	16	28,75	22,2	27,4	22	-0,3	16	16
07:45	18,25	-8	16	28,25	24	27,4	24,75	-0,4	16	16
08:00	19,5	-8	16	27	26,8	27,4	20	-0,5	16	16
08:15	17,5	-8	16	28,25	30,1	27,4	22,5	-0,4	16	16
08:30	18,5	-8	16	25,75	31,3	27,4	19,75	-0,5	16	16
08:45	18	-8	16	27	32	27,4	19,75	-0,5	16	16
09:00	16,25	-8	16	25,5	32,8	27,4	19,5	-0,3	16	16
09:15	18,5	-7,9	16	26	33,4	27,4	19,5	0	16	16
09:30	16,25	-7,5	16	25,25	33,8	27,4	19,75	0,4	16	16
09:45	16,75	-6,4	16	25,5	34,9	27,4	18,5	0,5	16	16
10:00	18	-4,8	16	26,25	35,3	27,4	19,25	0,5	16	16
10:15	17,25	-3,4	16	26,5	36,1	27,4	18	0,6	16	16
10:30	16,5	-2,7	16	24,75	36,7	27,4	18	1,2	16	16
10:45	17,25	-2,8	16	24,5	37	27,4	18,25	1,5	16	16
11:00	16	-2,9	16	25,75	37,3	27,4	18	1,9	16	16
11:15	16,75	-3,2	16	25,75	37,5	27,4	18,75	2,4	16	16
11:30	17,75	-4,1	16	27	36,3	27,4	18	2,9	16	16
11:45	17	-4	16	25	37,5	27,4	18,75	3,4	16	16
12:00	17,25	-3,7	16	27,25	37,8	27,4	18,5	3,9	16	16
12:15	15,5	-3,5	16	27,5	37,5	27,4	19	3,2	16	16
12:30	17,75	-3,2	16	26,75	37,6	27,4	19	3,3	16	16
12:45	17,5	-3,1	16	25,25	35,4	27,4	17,75	3,6	16	16
13:00	17	-2,8	16	28,5	34,1	27,4	18,25	4	16	16
13:15	17	-2,5	16	25,75	34,2	27,4	17,5	4,4	16	16
13:30	19,5	-2,5	16	28,75	34,2	27,4	19	4,7	16	16
13:45	18	-2,5	16	27	34,1	27,4	20,5	5,1	16	16
14:00	18,5	-2,5	16	28,75	34,2	27,4	19	5,2	16	16
14:15	17,25	-2,4	16	30,5	34,4	27,4	21	5,4	16	16
14:30	18,75	-2,2	16	27,75	34,5	27,4	19,25	5,5	16	16
14:45	17,75	-2,2	16	28,5	34,5	27,4	19	5,7	16	16
15:00	16	-2,1	16	26,75	34,2	27,4	17,25	5,9	16	16
15:15	18,25	-2,1	16	27,5	34,3	27,4	18,75	5,9	16	16
15:30	16,75	-2,1	16	28,5	34,4	27,4	19	6,1	16	16
15:45	18,5	-2	16	28	34,4	27,4	17,5	6	16	16

Filiale 2										
Zeit	24.1.			16.7.			22.10.			Modell alt kWh
	real kWh	Temp °C	Modell neu kWh	real kWh	Temp °C	Modell neu kWh	real kWh	Temp °C	Modell neu kWh	
16:00	17,75	-2,1	16	30	34,4	27,4	18,5	6,1	16	16
16:15	17,25	-2,4	16	28,5	34,3	27,4	18,38	6,1	16	16
16:30	18,25	-2,2	16	28,5	34	27,4	18,38	6,1	16	16
16:45	17,25	-2	16	28	34,1	27,4	18,25	6,1	16	16
17:00	17,25	-1,9	16	29,75	34	27,4	18,75	5,8	16	16
17:15	18,25	-2,1	16	29	33,9	27,4	19	5,9	16	16
17:30	17,25	-2	16	28	33,7	27,4	17	5,6	16	16
17:45	16,5	-2	16	27,5	33,8	27,4	19,5	5,3	16	16
18:00	17,75	-2	16	27,25	33,6	27,4	18	5	16	16
18:15	18,25	-2,7	16	28,75	33,4	27,4	18,5	4,6	16	16
18:30	16,75	-3	16	26,75	33,2	27,4	17,5	4,4	16	16
18:45	17	-3,2	20	26,25	33	31,4	17,75	4,1	20	16
19:00	16,75	-3,1	20	26,75	32,9	31,4	18,5	3,8	20	16
19:15	17,5	-2,9	20	27,5	32,5	31,4	17,5	3,5	20	16
19:30	19	-2,9	20	28	32,1	31,4	17,75	3,3	20	16
19:45	18,75	-3	20	27,25	31,6	31,4	20,25	2,9	20	16
20:00	18,5	-3,1	20	27	31,2	31,4	18,75	2,6	20	16
20:15	16,5	-3,3	20	27	30,9	31,4	19,75	2,3	20	16
20:30	17,75	-3,2	20	28,25	30,1	31,4	18,5	1,9	20	16
20:45	16,75	-3,1	10	28,5	29,7	21,4	19	1,7	10	16
21:00	23,25	-3	10	26,5	29,2	21,4	19	1,8	10	16
21:15	25,75	-2,7	10	28,25	28,5	21,4	18	1,7	10	16
21:30	19	-2,4	10	30,25	28,5	21,4	21,5	1,4	10	16
21:45	18,75	-2,4	10	30,25	28,1	21,4	23,5	1,2	10	16
22:00	23	-2,5	10	33,25	27,8	21,4	22	1	10	16
22:15	25,25	-2,8	10	37,5	27,5	21,4	26,75	0,8	10	16
22:30	23,75	-2,6	10	35,75	27,2	21,4	29,75	0,7	10	10
22:45	21,75	-2,6	10	30,25	26,8	21,4	25,75	0,5	10	10
23:00	20	-2,5	10	29,25	26,4	21,4	21,25	1	10	10
23:15	18,25	-2,3	10	30	26,2	21,4	22	1,3	10	10
23:30	19	-2,1	10	28,25	26	21,4	22	1,4	10	10
23:45	17,5	-2	10	25	25,7	21,4	19,5	1,6	10	10

Literaturverzeichnis

- [BAM02] Günter Bamberg und Franz Baur: „Statistik“, 12. überarbeitete Auflage, ISBN 3-486-27218-7, Oldenburg, München, Wien, 2002
- [BAN92] Hans Bandemer und Wolfgang Näther: „Fuzzy data analysis“, ISBN 0-7923-1772-6, Kluwer, Dodrecht, 1992
- [CHO00] A. A. Chowdhury und Don O. Koval: „Development of system reliability performance benchmarks“, in *IEEE Transactions on Industry Applications*, ISSN: 0093-9994, 36(3), S. 899-903, Mai-Juni 2000
- [EL08] Deutsche Energie Agentur (2008), [Online], Verfügbar unter <http://www.stromeffizienz.de/eu-label.html> [abgerufen am: 10.5.2008]
- [ESC07] Wien Energie Vertrieb GmbH & CoKG (2007), [Online], Verfügbar unter: <http://www.energiesparcheck.at> [abgerufen am: 10.5.2008]
- [FLA99] Patrick Flandrin: „Time-frequency, time-scale analysis“, Translated from the French by Joachim Stöckler, ISBN 0-12-259870-9, Academic Press, San Diego, California, 1999
- [FRA05] Barbara Schlomann (Projektleitung) et al.: „Technische und rechtliche Anwendungsmöglichkeiten einer verpflichtenden Kennzeichnung des Leerlaufverbrauchs strombetriebener Haushalts- und Bürogeräte“, Abschlussbericht an das Bundesministerium für Wirtschaft und Arbeit, 25. April 2005, [Online], Verfügbar unter: <http://www.isi.fraunhofer.de> [abgerufen am: 11.4. 2008]
- [HAM04] Broschüre: „Tipps zum Energiesparen für den Fleischer ‘Mustermann‘“, 2004, [Online], Verfügbar unter: <http://fhh.hamburg.de> [abgerufen am: 11.4. 2008]
- [HAN02] David J. Hand: „Pattern detection and discovery“, ISBN 3-540-44148-4, Springer, Berlin, 2002
- [IEEE96] IEEE Task Force on Bulk Power System Reliability: „Reporting bulk power system delivery point reliability“, in *IEEE Transactions on Power System*, ISSN: 1558-0679, 11, S. 1262-1267, August 1996
- [LEV05] Markus Levy: „Evaluating digital entertainment system performance“, in *Computer*, ISSN: 0018-9162, 38(7), S. 68-72, Juli 2007
- [LIN99] Run-Bin Lin, Isaac Shuo-Hsiu Chou, Chi-Ming Tsai: „Benchmark circuits improve the quality of a standard cell library“, in *Proceedings of the ASP-DAC '99. Asia and South Pacific Design Automation Conference, 1999.*, ISBN: 0-7803-5012-X, 1, S. 173-176, Jänner 1999

- [LO04] K. L. Lo, Zuhania Zakaria: „Electricity consumer classification using artificial intelligence”, in *39th International Universities Power Engineering Conference, 2004. UPEC 2004.*, ISBN: 1-86043-365-0, 1, S. 443-447, September 2004
- [MEI05] Stephan Meier, Philip C. Kjaer: „Benchmark of Annual Energy Production for Different Wind Farm Topologies”, in *IEEE 36th Power Electronics Specialists Conference, 2005. PESC '05.*, ISBN: 0-7803-9033-4, S. 2073-2080, 2005
- [MIY88] Misao Miyata, Hidechika Kishigami, Kosei Okamoto, Shigeo Kamiya: „The TX1 32-bit microprocessor: performance analysis, and debugging support”, in *IEEE Micro*, ISSN: 0272-1732, 8(2), S. 37-46, April 1988
- [PAL05] Peter Palensky: „The JEVIS Service Platform - Distributed Energy Data Acquisition and Management”, in *The Industrial Information Technology Handbook*, S. 111-121, CRC Press, 2005
- [PAL01] Georgios Paliouras: „Machine learning and its applications”, ISBN 3-540-42490-3, Springer, Berlin, 2001
- [PRE94] Adalbert Prechtl: „Vorlesungen über die Grundlagen der Elektrotechnik, Band 1“, 1. Auflage, ISBN 3-211-82553-3, Springer-Verlag, Wien, New York, 1994
- [SIE93] Daniel P. Siewiorek, John J. Hudak, Byung-Hoon Suh, Zary Segall: „Development of a benchmark to measure system robustness”, in *The Twenty-Third International Symposium on Fault-Tolerant Computing, 1993. FTCS-23. Digest of Papers.*, ISBN: 0-8186-3680-7, S. 88-97, Toulouse, 1993
- [STR00] Dirk Stroobandt, Peter Verplaetse, Jan Van Campenhout: „Generating synthetic benchmark circuits for evaluating CAD tools”, in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, ISSN: 0278-0070, 19(9), S. 1011-1022, September 2000
- [TUD07] Darcian Tudor, Marius Marcu: „A Power Benchmark Experiment for Battery-powered Devices”, in *4th IEEE Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 2007. IDAACS 2007.*, ISBN: 978-1-4244-1347-8, S. 91-95, September 2007
- [WEI90] Reinhold P. Weicker: „An overview of common benchmarks”, in *Computer*, ISSN: 0018-9162, 23(12), S. 65-72, Dezember 1990
- [WEI99] Alan R. Weiss: „The standardization of embedded benchmarking: pitfalls and opportunities”, in *(ICCD '99) International Conference on Computer Design, 1999.*, ISBN: 0-7695-0406-X, S. 492-508, Oktober 1999
- [ZHO05] Qiong Zhou, Janusz W. Bialek: „Approximate model of European interconnected system as a benchmark system to study effects of cross-border trades”, in *IEEE Transactions on Power Systems*, ISSN: 1558-0679, 20(2), S. 782-788, Mai 2005