DATA ANALYTICS OF SMART GRID ENVIRONMENT FOR EFFICIENT MANAGEMENT OF DEMAND RESPONSE

A Thesis submitted in fulfillment of the requirement for the award of the degree of

DOCTOR OF PHILOSOPHY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by:

Anish Jindal

(Registration No: 901403003)

Under the guidance of:

Dr. Neeraj Kumar

Associate Professor, CSED

Dr. Mukesh Singh

Associate Professor, EIED

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY,

PATIALA – 147004

SEPTEMBER 2018

CERTIFICATE

I, Anish Jindal, Regn. No. 901403003, hereby declare that the thesis entitled **"Data Analytics of Smart Grid Environment for Efficient Management of Demand Response"** submitted to the Computer Science and Engineering Department at Thapar Institute of Engineering & Technology, Patiala, Punjab, India is an authenticated record of my own work for the award of the degree of "Doctor of Philosophy" under the supervision of Dr. Neeraj Kumar and Dr. Mukesh Singh. This report has not been submitted to any other institution for award of any other degree.

Anish Jindal

Regn. No. 901403003

Place: Patiala

Date: 18-09-2018

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

Verified by:

Dr. Neeraj Kumar
Associate Professor
CSED
Thapar Institute of Engg. & Tech.

Dr. Mukesh Singh
Associate Professor
EIED
Thapar Institute of Engg. & Tech.

ABSTRACT

The future of the power industry heavily relies on the use of modern electric grids integrated with information and communication technology (ICT). Such grids are commonly known as smart grids. The advantage of using smart grids is that they provide a better quality of service in terms of better resource and asset management, detecting faults in the system, efficient energy consumption by reducing the demand and supply gap, and peak load reduction. Data analytics has already been applied extensively in the power sector to provide various services such as-demand forecasting, revenue protection, and data visualization. However, there are still many areas which can be benefited by using data analytical techniques. One such area is the demand response management in the smart grid environment where data analytics can be effective in order to manage the overall load on the grid. The entities involved in the smart grid comprise of power generation units, transmission and distribution units, and end-users. The end-users may belong to the different sectors such as–commercial, residential and transportation. The consumption data related to these users can be analyzed to provide many ancillary services in the smart grid and to improve the overall quality of service for the users. Keeping this in mind, the major focus of this thesis is on data analytics in the smart grid environment along with the demand response management of the connected loads. To achieve these tasks, four different schemes have been proposed in this thesis with an emphasis on data analytics and demand response management in smart grid. In the first technique, a top-down approach is designed to detect electricity theft in the power network based on decision tree (DT) and support vector machine (SVM).

Unlike the existing schemes, the proposed scheme detects and locates the real-time electricity theft at every level in power transmission and distribution (T&D). In the T&D level, the data received from various sensors is analyzed to detect the theft in the power lines by compensating for the T&D losses. At the consumer level, the DT is used to compute the value of expected load consumption in the smart homes which along with other attributes is given as an input to train the SVM classifier. Based on the training, the SVM detects the electricity theft at the consumer level using the input parameters received from various smart homes. The results obtained using the proposed scheme indicate that it detects the theft with high accuracy (i.e., 92.5%) and low false positives (i.e., 5.12%). The second scheme proposed in this thesis is designed for reducing the demand and supply gap in the grid by managing the demand response of smart homes and plug-in hybrid electric vehicles (PHEVs). For this purpose, the SVM classifiers are used to identify the users (smart homes or PHEVs), whose load profile needs to be regulated. The proposed scheme is a hierarchical scheme which manages the load profile of the grid in two phases. In the first phase, the residential loads (comprising of smart homes) are identified and managed according to the grid requirements. In the next phase, the charging rates of PHEVs are regulated when the residential loads are not sufficient to flatten the load profile of the grid. The results obtained using the case study of PJM and Open Energy Information dataset prove that the proposed scheme is effective in balancing the overall load profile of the smart grid. In the third scheme, peak load on the grid is reduced by analyzing the energy consumption patterns in the smart homes. Various factors are computed from the data gathered from the smart homes; based on which, algorithms for taking the demand response decisions to manage the consumer load profiles are proposed in the peak load scenario. The proposed scheme keeps the value of load curtailment in a smart home below the user specified curtailment value. However, in the cases where load curtailment is more than the user-specified value, incentives are provided to the consumers to compensate for the violation of consumer comfort. The efficacy of the proposed scheme has been validated using two case studies of realistic and harsh scenarios of electricity usage in smart homes. The results depict that the proposed scheme efficiently brings down the peak load demand by a factor of more than 27% when the proposed algorithms are used in tandem. Moreover, the proposed scheme also increases the savings of the consumers by reducing their overall electricity bills. The fourth scheme uses the tensor-based operations to reduce the data dimensionality of the data gathered from the power network. The proposed approach includes the extraction of core data from the gathered data by using tensor operations such as-matricization, vectorization and tensorization with the help of higher-order singular value decomposition. The core data is then used for various purposes such as managing the demand response of the loads in a smart city. The obtained results give a clear indication that the tensor-based approach in reducing the data dimensionality is effective and can be used for processing the gathered data in order to manage the demand response in the smart grid environment.

ACKNOWLEDGMENTS

First and foremost, I would like to thank the almighty God who gave me strength and courage to overcome all the obstacles and complete this endeavor. The successful completion of any task would be incomplete without acknowledging the people who made it possible. I would like to take this opportunity to express my gratitude to all those who made this journey possible. Words are often too less to express one's deepest regards, but lets give it a go.

I offer my sincerest gratitude to my supervisors, Dr. Neeraj Kumar (Associate Professor, CSED) and Dr. Mukesh Singh (Associate Professor, EIED), who have supported me throughout my Ph.D. work with their patience and knowledge; while providing me the room to work in my own way. I had the pleasure to visit USA and France with each of them and one simply could not have wished for better and friendlier supervisors. They led me to the correct direction at every stage of this research work. Apart from providing me with excellent supervision, strong cooperation and constant encouragement throughout this journey, they also shared their invaluable experiences with me to succeed in life. They have truly been the source of real inspiration for me and I will always remain indebted to them.

I am also grateful to the head of department, Prof. Maninder Singh, the former head, Prof. Deepak Garg, Ph.D. Coordinator, Dr. Parteek Bhatia and members of my doctoral committee, Prof. Inderveer Chana, Dr. Prashant Singh Rana and Prof. Ravinder Agarwal for their constructive suggestions and ensuring the correct pace of my work. I sincerely thank all the faculty and support staff of Computer Science and Engineering Department as well for their constant help whenever required. I am also obliged to the Director, Prof. Prakash Gopalan, Dean (RSP), Prof. O. P.

Pandey and the management of Thapar Institute of Engineering and Technology, who provided me with all the necessary resources and facilities to complete my work.

The chain of my gratitude will definitely be incomplete if I forget to thank my parents, Dr. Ashok Kumar Jindal and Dr. Anita Jindal, for their unconditional love, support and encouragement in every phase of my life. It was due to my father's confidence and vision that I started my Ph.D. in the first place. Since then, the journey of Ph.D. has been a sweet and bitter ride at times which leads to a special mention for my mother who stood by me through thick and thin, and gave me courage at the times when I felt really low. Her constant motivation showed me the silver lining in the dark clouds. I would also like to express heartfelt thanks to my grandfather who always believed in me and whose blessings have truly played the role of game changer in my life. It would also be unworthy of me if I forget to mention about the heavenly blessings constantly showered on me by my grandmother. I would also like to pay my sincere regards to all my relatives and cousins for their constant motivation and support. They made this journey easier with words of encouragement which helped me in finishing my work.

I would also like to thank my friends and colleagues with whom I have traveled this journey of research. A special thanks to my research group, Dr. Amit Dua and budding doctors, Mr. Gagangeet Singh Aujla, Ms. Kuljeet Kaur, Mr. Rajat Chaudhary, Mr. Karanveer Dhingra, Mr. Ishan Budhiraja, Ms. Aaisha Makkar, Ms. Shubhani Aggarwal, Ms. Sumedha Sharma and Ms. Rubi Rana. These people have made my research journey all the more memorable and pleasant.

As one cannot mention the names of all well-wishers, friends and beloved ones, I would like to pay my regards to one and all who supported me during this journey of knowledge. The financial help from Council of Scientific and Industrial Research, New Delhi under grant no. 09/677(0025)/2015-EMR-I is duly acknowledged which has provided me with the opportunity to work without worrying about the funds.

(**Anish Jindal**)

List of Publications

Journal Publications (SCI/SCIE):

1. Anish Jindal, Amit Dua, Kuljeet Kaur, Mukesh Singh, Neeraj Kumar, and S. Mishra, "Decision Tree and SVM-based Data Analytics for Theft Detection in Smart Grid," ***IEEE Transactions on Industrial Informatics***, vol. 12, no. 3, pp. 1005-1016, 2016. (SCIE, IF-5.430)

2. Anish Jindal, Neeraj Kumar, and Mukesh Singh, "Internet of Energy-based Demand Response Management Scheme for Smart Homes and PHEVs Using SVM," ***Future Generation Computer Systems,*** 2018, DOI: 10.1016/j.future.2018.04.003. (SCIE, IF- 4.639)

3. Anish Jindal, Mukesh Singh, and Neeraj Kumar, "Consumption-Aware Data Analytical Demand Response Scheme for Peak Load Reduction in Smart Grid," ***IEEE Transactions on Industrial Electronics,*** vol. 65, no. 11, pp. 8993-9004, 2018. (SCI, IF-7.05)

4. Anish Jindal, Neeraj Kumar, and Mukesh Singh, "A unified framework for big data acquisition, storage and analytics for demand response management in smart cities," ***Future Generation Computer Systems***, 2018, DOI: 10.1016/j.future.2018.02.039. (SCIE, IF- 4.639)

Conference Publications:

1. Anish Jindal, Neeraj Kumar, and Mukesh Singh, "A Data Analytical Approach Using Support Vector Machine for Demand Response Management in Smart Grid," in ***IEEE PES General Meeting***, Boston, MA, Jul. 2016, pp. 1-5.

Contents

List of Figures

List of Tables

LIST OF IMPORTANT ABBREVIATIONS

AAF	Appliance adjustment factor
ACP	Appliance curtailment priority
AMI	Advanced metering infrastructure
ANN	Artificial neural network
AP	Access point
API	Appliance priority index
ARF	Appliance run-time factor
AUF	Appliance usage factor
CC-DADR	Consumer centric data analytical demand response
CCF	Consumer curtailment factor
CS	Charging station
DADR	Data analytical demand response
DER	Distributed energy resources
DT	Decision tree
EV	Electric vehicle
FCP	Fast chargeable plug-in hybrid electric vehicle
HEMS	Home energy management system
HVAC	Heating, ventilation and air conditioning
ICT	Information and communication technologies
IoE	Internet of energy
LAF	Load adjustment factor
LIBSVM	Library for support vector machine

MILP	Mixed integer linear programming
MINLP	Mixed integer nonlinear programming
NCP	Normal chargeable plug-in hybrid electric vehicle
PHEV	Plug-in hybrid electric vehicle
PJM	Pennsylvania-New Jersey-Maryland
PMU	Phasor measurement unit
PV	Photo voltaic
RBF	Radial basis function
RCI	Relative consumption index
RES	Renewable energy sources
SCP	Slow chargeable plug-in hybrid electric vehicle
SoC	State of charge
SoH	State of health
SVD	singular value decomposition
SVM	Support vector machine
T&D	Transmission and distribution
UC-DADR	Utility centric data analytical demand response

Chapter 1

Introduction

With the advent of new technological breakthroughs, many concepts and techniques have evolved over the years. These technological advancements have revolutionized many application areas which have led to the smart city era. The key aspect of the smart city is to integrate information and communication technologies (ICT) for providing various services (such as–smart healthcare, smart connectivity, smart transportation, smart governance and public safety as depicted in Fig. 1.1) and to manage the resources (such as–water, energy, lighting and air quality control as shown in Fig. 1.2) optimally [4]. The benefits of smart cities have attracted many developing nations to adopt this concept in order to provide smart solutions for their citizens. The government of India is planning to develop 100 smart cities over the course of five years from 2015 to 2020 with each city set to receive the seed funds of 31 million dollars per year [5].

Out of the aforementioned services in a smart city, smart energy management which primarily focuses on utilization of the energy resources efficiently is a paramount concern. Energy is one of the most valuable resources of the modern era which needs to be utilized in such a manner so that there exists a balance between demand and supply. The mis-management of this valuable resource can create the energy gaps which can lead to power outages in some areas. These gaps may disrupt the normal functionality of the power grid whose main aim is to regulate the flow of power from

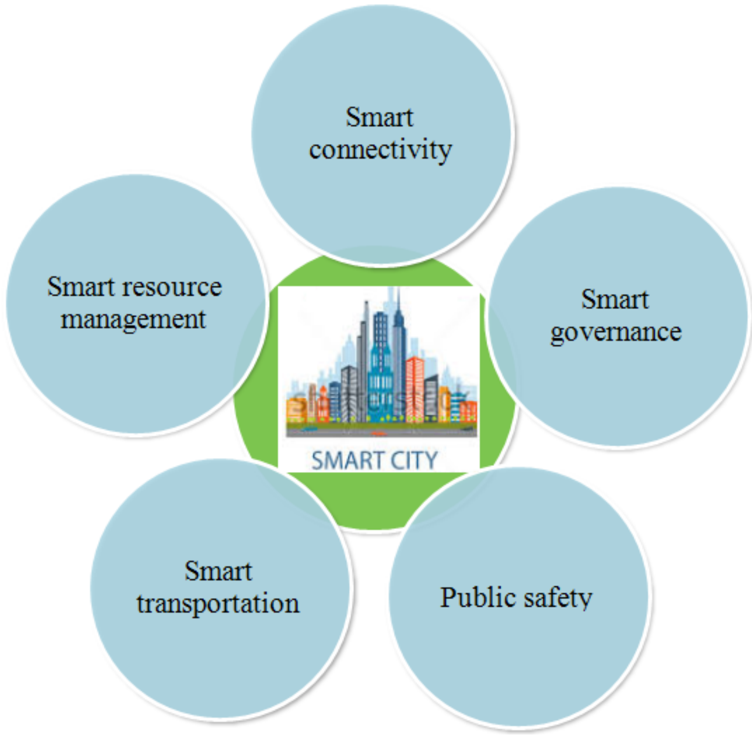


Figure 1.1: Application domains in a smart city.

the control center to the end-users. On the energy front, a smart city constitutes of various entities such as–electric vehicles, residential, industrial & commercial users, generation and transmission & distribution units. All these entities are connected with each another using ICT, thereby makes it a complex network of energy. The

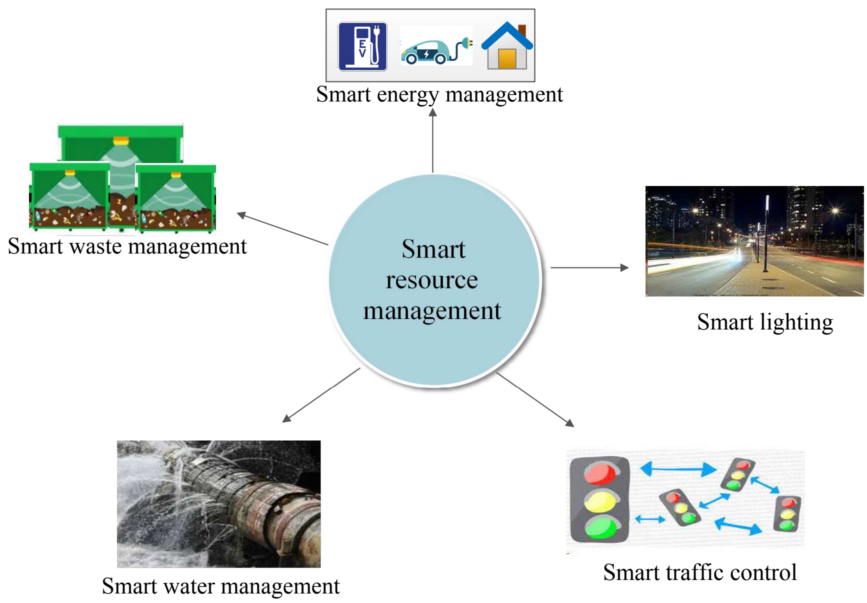


Figure 1.2: Smart resource management in a smart city.

complexity of this energy network can be handled in an efficient manner by using automated solutions at the grid. However, the rate at which the energy consumption is growing in the recent years is alarming. Fig. 1.3 depicts the predicted electricity growth usage for different sectors in the U.S. alone [1].

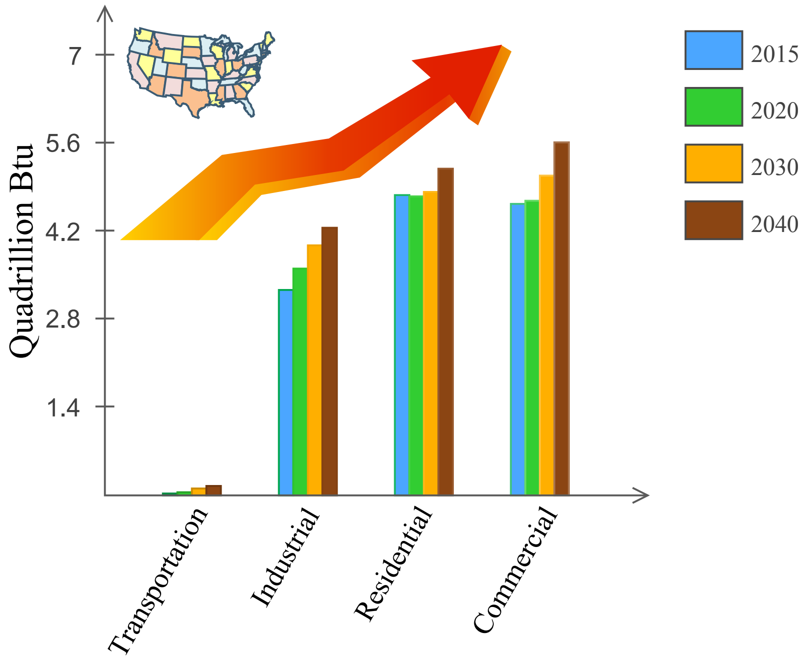


Figure 1.3: Predicted growth of electricity usage in different sectors [1].

To cater to the energy requirements in such fast-paced growing sectors, smart grid has emerged as one of the most powerful technologies. It can regulate the power flow in various phases such as–generation, distribution and consumption. The existing grids can be converted into smart grids by integrating with ICT which makes them capable of handling the load demand of the users automatically [6, 7].

1.1 Smart Grid

Smart grid technology is capable of managing the energy generated from various conventional or non-conventional sources in an efficient manner based upon the knowledge of demand and supply [8]. In smart grid, all the entities in the power flow (as shown in Fig. 1.4) from generation, transmission, distribution to the end-users generate the data related to power requirement and usage. This data can be analyzed in order to provide efficient energy management solutions to the end-users.

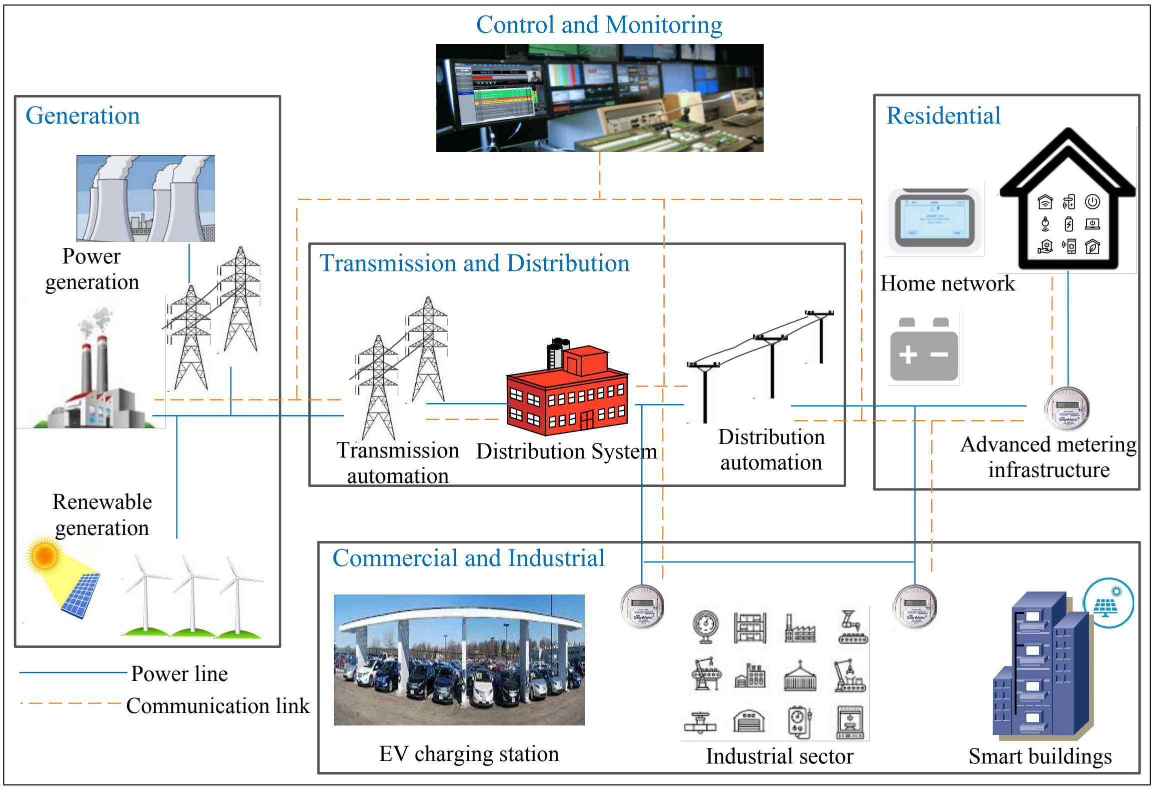


Figure 1.4: Data gathering in smart grid environment.

Moreover, in smart grid, utilities can predict energy requirements for the future based upon the consumption patterns of the end-users. Hence, by forecasting the load requirements of the consumers, the gap between demand and supply can be minimized by managing their demand response optimally [9]. demand response, in general, can be referred to as the changes made in the load demand of the users with respect to the policies adopted by the utility [10]. demand response management is important in the developing countries like ours to handle the load demand, where the energy generation resources are limited and the demand is increasing rapidly.

Micro-level of smart grid starts with the smart homes in the residential sector which are the major consumers of electricity as depicted in Fig. 1.3. Hence, managing their demand response can be fruitful so as to balance the overall load on the smart grid. If the smart home knows its present and future energy requirements, it can provide an energy consumption curve to the grid and accordingly, the grid can plan the distribution based on its available generation resources. In this way, smart grid can utilize its energy in a more efficient way as per the users' requirement. Once the energy is dissipated to various smart homes, they can control the appliances using home area networks (HANs) to balance the load as per the power supplied by the grid as shown in Fig. 1.4. Moreover, the distributed energy resources (DERs) such as PV panels can be utilized in smart homes or smart grids for managing the partial load demand to ease the burden on the smart grid [11, 12].

The management of power requirements by smart homes plays a crucial role in managing the overall load stability of the smart grids [13]. As the number of smart homes rises, thus data analytics needs to be performed in the smart grid domain for better resource utilization. Data analytics is a process which discovers new and meaningful information from the data sources. Data analytics can help to uncover interesting knowledge from the data gathered in the smart grid environment which might help to take efficient demand response decisions so as to manage the load profile of the grid. In smart grid environment, the data from smart homes can be gathered by using smart meters. The data from smart homes and smart grid is exchanged with the help of access points (APs) using an Internet source. These APs can be roadside units or wireless routers deployed at various locations to provide communication connectivity in the smart grid [14]. The data gathered from smart meters in the smart homes and other buildings are sent to the utility server using these APs. The data analytics can be performed at the utility server and the results are floated back to the smart grid/utilities (and other interested parties) for taking the intelligent decisions. The protocols used for various communication purposes with their details are mentioned in Table 1.1. In this table, *object* represents a smart home or other similar entity. Object-to-object, Object-to-access point and Access point-to-server communications signify short range, medium range and long range communications respectively. These protocols are used for collection of data from the smart homes or related entities to the utility server. Apart from these, the data in smart grid can also be collected from various other sources as mentioned in

Table 1.2.

Table 1.1: Various protocols used for the purpose of data communication.

Type	***Technology***	***Protocols***	***Frequency bands***	***Data rate***
Object-to-object	Passive RFID	IEEE 802.15.4f	915 MHz	<4 Mbps
	DSRC/WAVE	IEEE 802.11p	5.850-5.925 GHz	3-27 Mbps
	DSA	IEEE 802.11af	476-494 MHz	1 Mbps
Object-to-access point	WAVE	IEEE 802.11p	5.850-5.925 GHz	3-27 Mbps
Access point-to-server	Wi-Fi	IEEE 802.11 a/b/g	2.4 - 5 GHz	1-54 Mbps
	WiMAX	IEEE 802.16	1.25 - 20 MHz	30 Mbps - 1Gbps

RFID - Radio Frequency Identifier
DSRC - Dedicated short-range communication
WAVE - Wireless Access in Vehicular Environment
DSA - Dynamic Spectrum Access
WiMAX - Worldwide Interoperability for Microwave Access

Apart from smart homes, plug-in hybrid electric vehicles (PHEVs) also have a huge impact on the electricity market because of their increased popularity in transportation sector [15, 16]. According to the Global EV Outlook, there will be around 20 million electric vehicles (EVs) on road by the year 2020 [17]. These EVs can be leveraged to manage the overall load on smart grid by regulating their charging requirements [18]. These EVs/PHEVs can be used for peak shaving and valley filling when the grid suffers from load fluctuations in the peak and off-peak hours respectively [19]. Peak shaving is the technique to scale down the load profile of grid so as to comply with the available energy with the grid. Similarly, valley filling is used when load demand is less than the available energy in order to reduce the gap between demand and supply [20]. PHEVs which require charging can be scheduled for time instances in which load requirement is less, i.e., valley filling; and PHEVs that wants to discharge their energy in order to support grid stability are scheduled when load demand is more, i.e., peak shaving. Moreover, the charging rates of PHEVs can be controlled during peak and off-peak hours to manage their load requirements according to the grid constraints. For efficient peak shaving and

Table 1.2: Sources of data collection in smart grid.

Data type	**Technology involved**	**Remarks**
AMI	Smart meters	Data generated by the smart meters in the homes has grown manifolds since their inception.
Distribution automation	Grid equipment	Sensors deployed at various locations in the power network to sense the data values at very short intervals of time
Third-party	Off-grid data sets	To study the effect of dynamic pricing policies and use of renewable energy sources, utilities are integrating the data taken from other sources.
Asset management	Smart devices	To control and forecast the faults in grid equipment, a considerable amount of data is gathered from the smart devices.

valley filling, the data gathered from PHEVs can be helpful in predicting their contributions to the smart grid in advance.

However, the conventional techniques used in power grids are not sufficient to gather the required information. To overcome this drawback, power grids are made smarter all around the globe by integrating with advanced metering infrastructure (AMI) and other smart sensors [21]. By using these types of equipment, data collection has increased manifolds in contrast with tradition grid where data was collected monthly or bimonthly about the meter readings. Therefore, an unprecedented amount of data is being collected nowadays in smart grid with respect to generation and consumption of power [22]. By analyzing this data, better resource utilization can be done. According to the National Institute of Standards and Technology report, the benefits of modernization of grids are more than five times as compared to the one-time deployment cost [23]. According to the initial assessment by the American Council for an Energy-Efficient Economy, the use of ICT and smart appliances can save about $80 billion in America's annual bill [24]. This is possible only if the analytics on the gathered data is performed. Moreover, the use of AMI and intelligent supervisory control and data acquisition (SCADA) units is essential to improve the quality of gathered data for better analysis.

1.2 Data Analytics

Data analytics is generally classified into four broad categories; descriptive, diagnostic, predictive and prescriptive; each of which is defined as follows.

1. *Descriptive analytics* provides the information about what happened and helps in creating a plausible explanation for it. This analytics intends to find out the useful information from the underlying data which can be used for further processing.

2. *Diagnostic analytics* helps in understanding the reason behind the occurrence of a particular event and also helps to understand the system behavior by identifying various challenges and opportunities.

3. *Predictive analytics* performs probabilistic predictions to identify future trends on the basis of present information.

4. *Prescriptive analytics* is used to determine the outcome of the particular events on the basis of given data and devise plans to handle such events.

In the smart grid, these types of analytics are often applied in order to manage the resources and to improve the quality of service for the users [25]. The overall flow of data from data collection to take any decision in a data analytical process in the smart grid is shown in Fig. 1.5. In the first stage, the data is required to be gathered on which the analytics can be performed.

As discussed above, this data can be gathered from various sources. More precisely, these sources include:

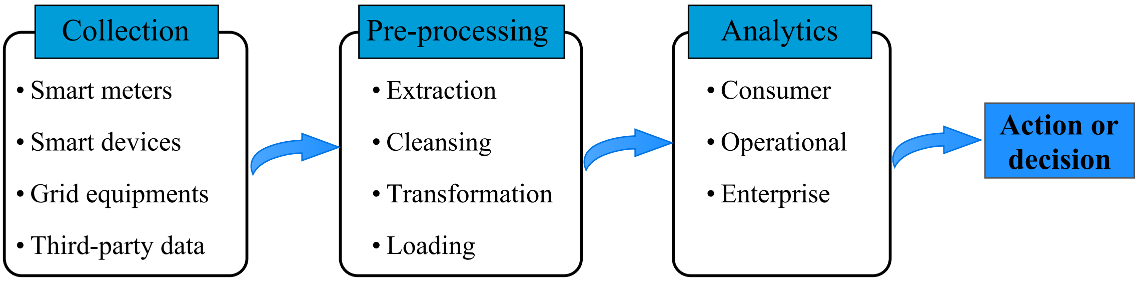


Figure 1.5: Flow in data analytics process.

- *Smart meters:* Due to a sudden rise in the deployment of smart meters in homes, data is generated by them at an expeditious rate.

- *Smart devices:* To control and forecast the faults in grid equipment, a considerable amount of data is generated by smart devices placed at various entities in smart grid environment.

- *Grid equipment:* Sensors deployed at various locations in the power network that sense the data values at very short intervals of time.

- *Third-party data:* Data can also be gathered by third parties to study the effect of various policies like dynamic pricing and use of renewable sources in homes for energy management.

Once the data has been collected, the next step is to pre-process the data so as to remove any inconsistencies from it. As the data is gathered from various sources, it may be in a variety of formats and may contain missing or erroneous values. These data values are then cleaned to remove erroneous values using the processes as described below [26].

- *Extraction:* It is reading the data from various sources which can be in a variety of formats.

- *Cleansing:* Data may contain some missing or erroneous values, so these values are either deleted or corrected which is called cleansing of data.

- *Transformation:* In this step, the data is converted from the current format into the target repository's format.

- *Loading:* Finally, the data is loaded into the repositories or warehouses where it is stored for further processing.

Now, from the viewpoint of smart grid, the data analytical techniques such as–consumer, operational and enterprise analytics are applied to the pre-processed data to extract the useful information from it [25].

- *Consumer analytics:* It is based on the data gathered from the consumer end. This type of analytics include energy forecasting, consumption analysis and theft detection.

- *Operational analytics:* It is performed based on the data gathered from sensors deployed in the smart grid or third party data. It includes asset maintenance, outage management and distribution optimization.

- *Enterprise analytics:* It is the analytics performed for real-time grid awareness, visualization of data, etc.

On the basis of information gained after performing these (or one of these) analytics, informed actions are taken (or decisions are made).

Data analytics helps in catering various issues related to smart grid in the past such as–demand forecasting [27], interactive visualization [28] and revenue protection [29]. Moreover, various commercial and pilot projects have been started by the companies which apply data analytics in various domains of smart grid as listed in Table 1.3. Thus, it is evident from all these studies and projects that data analytics can be applied intelligently to the data gathered from the smart grid environment to provide a robust demand response management solution.

Table 1.3: Various data analytical projects.

Project Name	**Purpose**
General Electric's GridIQ Insight platform [30]	Outage analytics
Siemens' eMeter [31]	Predictive analytics
AutoGrid DROMS [32]	Demand response analysis
Oracle Opower [33]	Peak load analysis
OGE power [34]	Customer planning
Edison's distribution SCADA project [35]	Grid awareness

1.3 Need of Data Analytics

The utilization of renewable energy sources (RES) and EVs to manage the partial home loads has made the load profile of the home users' variable. The sudden changes in the load demands can lead to the instability in the grids if these are not handled promptly. In the absence of analytical and computational support, the current infrastructure is not able to analyze and handle such dynamic changes in the usage patterns of the consumers [36]. As there are millions of users and thousands of control points which generate the data, so it becomes a complex data analytical problem to manage such a large number of incoming requests. This is evident from the fact that nearly 1 billion smart meters would be deployed throughout the world at the end of 2020 [36]. These devices generate an enormous amount of data which needs to be gathered and processed efficiently. By analyzing this data, power companies or utilities can minimize their losses and provide a better quality of service to the consumers. GTM research forecasted that the spending on data analytics by power utilities would go upto $20 billion by the end of this decade [37]. Data analytics can change the perception of both the consumers and utilities, the way they use the power. Data analytics plays a crucial role in managing the future needs of the customers and influences many areas in the smart grid environment.

Data analytics acts as a key player in the smart grid by analyzing the collected data. For instance, revenue protection or loss/theft detection is one such area where analytics can save hundreds of millions for the utilities because of the stealing and mis-management of power [38]. The analytics performed for providing demand response management can help to optimize the electric consumption of the users for reducing the burden on the smart grid. This analytics can forecast the energy demands and helps in scaling down the energy requirements at peak hours by providing benefits to the customers during off-peak hours. In the future, a better understanding of customer behavior can prove invaluable to the utilities. After considering all these aspects of data analytics, it can be inferred that an intelligent data handling technique is the need of the hour as it is important to analyze and manage the data generated in the smart grid environment.

1.4 Impact of Data Analytics

The impact of performing data analytics in smart grid includes–increased customer satisfaction, better expenditure of capital, improved efficiency and improved reliability. Customer satisfaction is increased by gaining insights into customer experiences from the gathered data. The personalized solutions can be provided to the customers according to their usage pattern. The expenditure on capital can be optimized by reducing economic disparity in the grids and reducing faults in the system. The efficiency and reliability of the grid can be improved by forecasting the energy usage of consumers and managing the resources accordingly. In general, data analytics influences the smart grid and consumers in the following ways.

- **Operational efficiency:** It helps the smart grid to increase its operational efficiency by providing theft, loss and outage detection, and thus allowing better asset management.

- **Implementing strategies:** The data analytics about usage patterns of the consumers in the peak hours can help in saving money by shifting their loads to times where the price is less. This, in turn, alleviates the need for building new infrastructure by the utilities for generating more power in the peak hours.

- **Developing new models:** New business models can be developed to leverage the customers depending on the data analytics of demand response, customer behavior and power generation.

- **Improving grid resilience and load management:** Applying analytics on the gathered data allows the utilities to identify and rectify problems in the smart grid quickly and prevent power outages in the system. This also helps in providing better load management to the consumers for managing their energy use.

- **Customer satisfaction:** Data analytics also helps in providing customized plans for the consumers which maximize their benefits and increase their satisfaction.

1.5 Challenges and Constraints

Following are the major challenges that need to be addressed in order to manage the demand response and to perform data analytics on the data generated in smart grid environment.

i. **Data sharing, security and privacy:** A huge amount of data is available with retail energy providers and energy managers (like home security, improvement and telecommunication providers). This data is not shared with one another unless it is in the economic interest of all the parties. The customer's data security and privacy should also be maintained so that the customer feels safe to share its data with these vendors.

ii. **AMI data integration:** Utilities need to capture the data generated by the AMI. This data needs to be integrated with other parts of utility enterprise like customer systems and operation system which has its own challenges. This data can help in leveraging the customers by designing personalized plans for them based on their preferences.

iii. **Analytics on demand response:** The demand response analytics studies the customer behavior in various scenarios. The problem in performing this analytics is the variability in the response of the customers. Customer usage pattern depends on many factors like electricity price, user preferences and weather conditions. These factors must be taken into account while managing and analyzing the demand response.

iv. **Infrastructure requirements:** The data generated by the consumers is huge and changes with time, thus it is difficult to analyze. As the homes and vehicles have limited computational resources, the data needs to be stored and processed on the scalable infrastructure.

v. **Dynamic change management:** The dynamic change in consumers requirement must be incorporated with the smart grids so as to prevent the load fluctuations in it. Although, one change may not affect the grid stability, however, if these changes are happening with a large number of customers, then it can lead to grid breakdown.

vi. **Real-time grid awareness:** The utilization of solar PV, PHEVs and other DERs influences the grid stability due to their intermittent nature. They create extra pressure on the distribution systems. Thus, managing these resources in real-time is important for the grid stability.

viii. **Lack of standards for communication interoperability:** The requirement of a robust and real-time data communication standard still persists for both real-time communication and application interfacing for seamless data exchange between generation, transmission, distribution and customer applications [23].

ix. No unified place to access and implement the strategies on the data.

1.6 Summary of Research Questions

From the above discussion, it is clear that data analytics can help to address various issues in the power industry. However, to validate this claim, the following research questions are addressed in this thesis.

1. Whether data analytics can be applied to the data collected from the smart grid environment to mitigate various issues such as theft detection and demand response management?

2. Can the re-scheduling of smart appliances in smart homes be an effective way to manage the demand response in the smart grid?

3. Whether the management of electric vehicles can help in reducing the overall burden on the grid?

4. Can the load in smart homes be managed in such way so as to reduce the load demand on the smart grid without affecting the user comfort?

5. Can the process of handling the data gathered in the smart grid environment be streamlined in such a way that the storage space and processing time is reduced?

1.7 Thesis Organization

To answer these questions, the thesis has been organized in the following chapters:

- **Chapter 2: Preliminaries**

  This chapter provides the information of preliminary research works in the areas of demand response, data analytics and energy management in the smart grid. Moreover, the relative comparison of existing approaches is given in this chapter to chalk out relative advantages and disadvantages of these approaches in terms of various evaluation parameters.

- **Chapter 3: Data Analytics for Theft Detection**

  To answer the first research question, a scheme for detecting the electricity theft in the smart grid on the basis of data analytics is described in this chapter. The proposed scheme is primarily focused on the detection of non-technical losses particularly due to electricity theft. To detect the large-scale consumption of electricity in a fraudulent manner, this chapter demonstrates a comprehensive top-down scheme based on the combination of decision tree and support vector machine to precisely detect these thefts in the complex power networks. Furthermore, the obtained results indicate that the proposed scheme successfully detects the theft on the basis of data analytics and also reduces false positives to a great extent.

- **Chapter 4: Demand Response Management**

  This chapter addresses the first three research questions. The first one of which is related to the application of data analytics for managing demand response, other one is related to the load re-scheduled of smart appliances and finally, the electric vehicles are also used in load management at the smart grid. The proposed scheme scheme is divided into two hierarchical stages for flattening the overall load profile of grid. In the first stage, the residential and PHEV users are identified whose demands can be regulated. In the next stage, load in smart homes is curtailed on the basis of a pre-defined rule-base; whereas PHEVs are managed by controlling their charging rates. The simulation results on PJM benchmark data and Open Energy Information dataset prove that the proposed scheme is effective in maintaining the load profile of smart grid. Moreover, the results prove that the participation of EVs helps in reducing the overall burden on the grid.

- **Chapter 5: Peak Load Reduction**

  The fourth research question is answered by designing a scheme which manages the demand response of smart homes without affecting the user comfort for peak load reduction in the smart grid. The proposed scheme is primarily based on the analysis of consumers' consumption data gathered from smart homes for which factors such as–appliance adjustment factor, appliance priority index, appliance curtailment priority, etc. have been designed. Based on these factors, different algorithms with respect to consumer's and utility's perspective have been proposed to take demand response decisions in the peak load scenario. In addition to it, an incentive scheme has also been presented to increase the consumers' participation in the proposed scheme. The results obtained show that it efficiently reduces the peak load at the grid to a great extent. Moreover, it also increases the savings of the consumers by reducing their overall electricity bills.

- **Chapter 6: Data Processing and Management**

  In this chapter, the last research question is tackled by presenting a tensor-based data management technique to reduce the dimensionality of the gathered data. The application of this scheme is demonstrated on the data gathered from the Internet-of-Energy (IoE) environment in a smart city. The core data is extracted from the gathered data by using tensor operations such as–matricization, vectorization and tensorization with the help of higher-order singular value decomposition. After reducing the dimensionality of data, it can be used for providing many services in smart cities and its application to provide demand response services in a smart city is discussed in this chapter. The results obtained clearly indicate the supremacy of the proposed tensor-based scheme over the traditional scheme to classify the end-users.

- **Chapter 7: Conclusion and Future Scope**

  This chapter concludes the thesis by highlighting the contributions made towards the proposed research domains. Moreover, this chapter also provides future directions of the work carried out in this thesis.

Chapter 2

Preliminaries

This chapter provides the information of preliminary research works in the domain areas of data analytics, demand response, data analytical demand response and energy management in the smart grid[1]. Moreover, the relative comparison of existing approaches is given in this chapter for analyzing the advantages and disadvantages of these approaches using various evaluation parameters. To summarize the overall contribution of this chapter, exiting proposals in the literature have been classified into various categories as shown in Fig. 2.1. The first category gives an overview of various schemes to utilize the data analytics solutions in smart grid. The second category, i.e., demand response management in smart grid focuses on the schemes

[1]The contents of this chapter are partly published in:

- A. Jindal, A. Dua, K. Kaur, M. Singh, N. Kumar, and S. Mishra, “Decision Tree and SVM-based Data Analytics for Theft Detection in Smart Grid,” *IEEE Transactions on Industrial Informatics*, vol. 12, no. 3, pp. 1005-1016, 2016.
- A. Jindal, N. Kumar, and M. Singh, “Internet of Energy-based Demand Response Management Scheme for Smart Homes and PHEVs Using SVM,” *Future Generation Computer Systems*, 2018, DOI: 10.1016/j.future.2018.04.003.
- A. Jindal, N. Kumar, and M. Singh, “A Data Analytical Approach Using Support Vector Machine for Demand Response Management in Smart Grid,” in *IEEE PES General Meeting*, Boston, MA, Jul. 2016, pp. 1-5.
- A. Jindal, M. Singh, and N. Kumar, “Consumption-Aware Data Analytical Demand Response Scheme for Peak Load Reduction in Smart Grid,” *IEEE Transactions on Industrial Electronics,* vol. 65, no. 11, pp. 8993-9004, 2018.
- A. Jindal, N. Kumar, and M. Singh, “A unified framework for big data acquisition, storage and analytics for demand response management in smart cities,” *Future Generation Computer Systems*, 2018, DOI: 10.1016/j.future.2018.02.039.

which give more emphasis on the actual demand response management without giving much importance to the underlying technique used. Then, the schemes which use the concept of data analytics for providing the demand response management solutions are summarized in the third category; while the fourth category focuses on the home energy management systems for energy utilization in smart homes. Finally, the application area of dimensionality reduction of data in smart grid is discussed. Each classification of the existing schemes is described as follows.

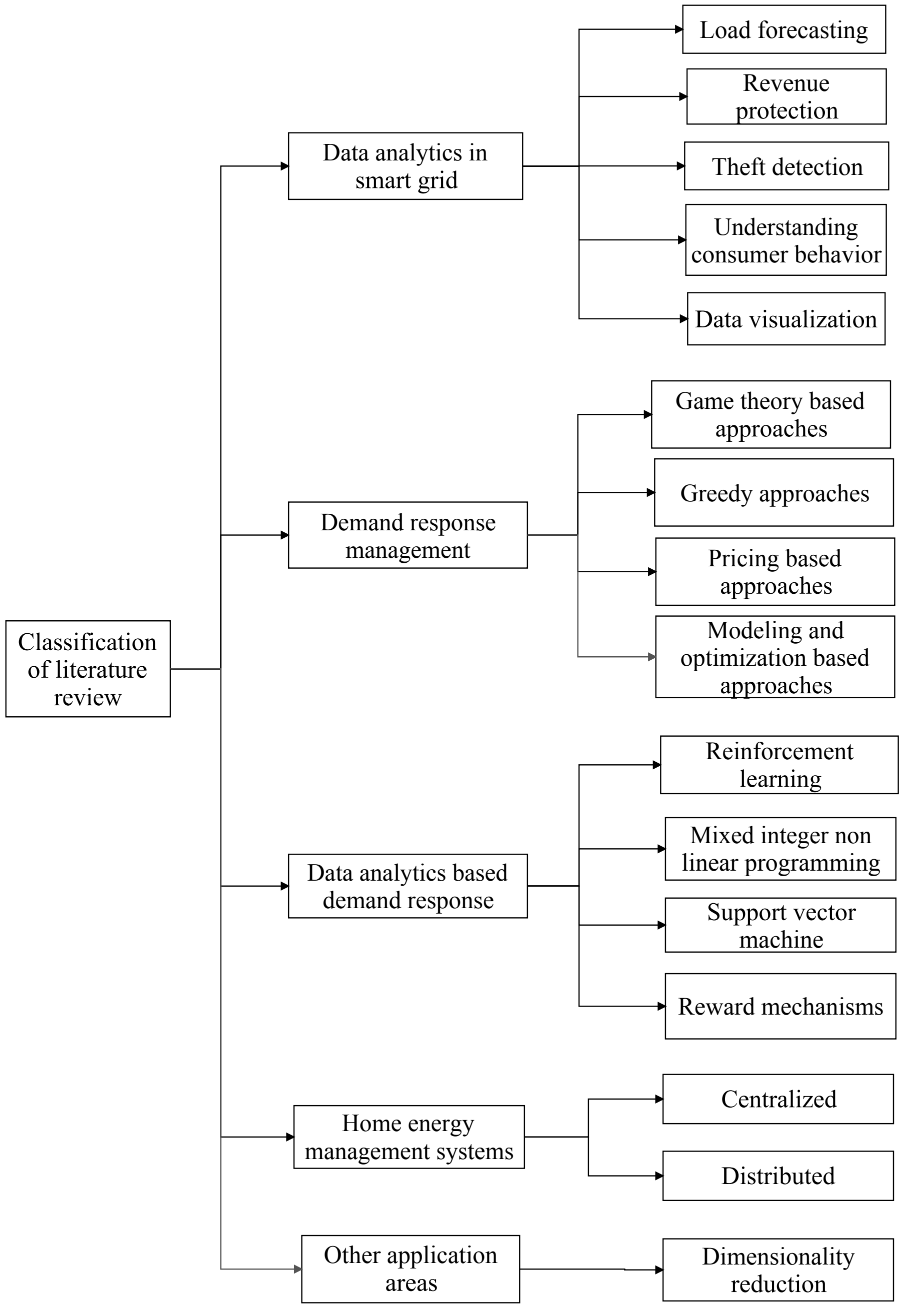


Figure 2.1: Classification of existing literature review.

2.1 Data Analytics in Smart Grid

Many analytical algorithms such as–regression, classification and clustering algorithms have been used in power sector to mitigate various issues related to smart grid. For example, Mirowsk *et al.* [27] analyzed the multi-year meter data to forecast the load demand in smart grid using data analytics. The authors developed a demand forecasting scheme to improve the overall accuracy and then they compared their model with various state-of-the-art short-term load forecasting schemes. Authors in [29] analyzed meter data by integrating cumulative sum and shewhart algorithms to identify the meters with irregular usage patterns for revenue protection. The authors preprocessed the data by normalizing the effect of temperature, analyzing unusual long missing or zero data and other outliers, and then applied these algorithms for change detection. Jokar *et al.* [39] used data analytics for electricity theft detection. The authors analyzed the electricity consumption patterns of the users and classified the users as normal or malicious by using support vector machine (SVM). Han *et al.* [40] used SVM to analyze the response of consumers under the various time of use electricity pricing policies. This analysis can further help the utilities to decide the optimal pricing policy in varying load demand scenarios.

Zhu *et al.* [28] proposed an exploratory tool to visualize the consumer data and provided the capabilities to the end users in order to interact with the power systems based on this visualization. The developed data-driven tool also helped the users to visualize the power system's configurations at different levels which could be used to plan the electricity usage in their homes. Wang *et al.* [41] used a differential evolution based SVM classifier to put forth an electricity price forecasting framework. For this purpose, the authors combined random forest algorithm and relief-F algorithm for hybrid feature selection on the basis of grey correlation analysis to remove duplicate features. After this step, principal component analysis was applied to extract the important features; on the basis of which, a differential evolution based SVM classifier was trained to accurately forecast the electricity prices.

Haben *et al.* [42] analyzed the smart meter data to make clusters of consumers for understanding their load demand and behavior. The four major time periods were identified for which the data was analyzed to form relevant attributes for clustering. Then, a finite mixture model was employed to cluster 10 distinct behavior groups attributed to various customers based on their demand and variability. All these aforementioned schemes are summarized in Table 2.1 in terms of their application area and a brief description. Moreover, Alahakoon and Yu [43] suggested various domain applications in energy systems where smart meter analytics can be helpful.

Table 2.1: Various data analytical techniques used in smart grid.

Scheme	**Application area**	**Description**
Mirowsk *et al.* [27]	Load forecasting	Developed a demand forecasting scheme that improved the overall forecasting accuracy.
Wu *et al.* [29]	Revenue protection	Identified the smart meters with irregular usage patterns for revenue protection.
Jokar *et al.* [39]	Theft detection	Analyzed the electricity consumption patterns of the users and classified the users as normal or malicious.
Han *et al.* [40]	Understanding consumer behavior	Analyzed the response of consumers under various time of use electricity pricing policies.
Zhu *et al.* [28]	Interactive visualization	An exploratory tool to visualize the consumer data and capabilities to the end users to interact with the power systems.
Wang *et al.* [41]	Price forecasting	Differential evolution based SVM classifier was developed to put forth an electricity price forecasting framework.
Haben *et al.* [42]	Understanding consumer behavior	Made clusters of consumers for understanding their load demand and behavior.

The researchers have also focused on regression and clustering algorithms to understand the consumer behavior and to model the working of various appliances. For example, authors in [44] studied the effect of outside air temperature on the regression models for load demand reduction. The authors made clusters using k-means clustering algorithm and then checked for the temperature sensitivity using various regression models. Zhang *et al.* [45] designed a baseline method to utilize the metered load to develop correlations between normal day loads and event loads. Initially, consumers' clusters were formed using k-means clustering and then, the baseline in each cluster was estimated using linear regression for different temperature settings. In [46], authors developed a short-term load forecasting scheme for studying the effect of demand response programs to the variation in user's consumption. The obtained results suggested that the users with more variable consumption were most likely to reduce their consumption in demand response scenario as compared to the less variable consumption users.

Hatton *et al.* [47] noted that the initial demand curves of the homes were a better indicator to select a control group for managing the demand response. For selecting the control groups, the authors used the constrained regression method to calculate the baselines. The major advantage of their method was that the control group could adapt to the changing number of participants in the demand response management process. In [48], the authors applied regression model to find out the response of the participants in the relative kW responses in BC Hydro's residential time-of-use pilot study. The authors noted that the proposed load control strategy in the peak days was a highly effective demand response management strategy during winters. Klaassen *et al.* [49] modeled the load profile of the smart washing machines by using stepwise multiple regression and analyzed the effects of an incentive based program for shifting the load demand of the washing machines. The authors noted that the load model was improved significantly when the PV generation was used to manage the load demand. The authors in [50] compared various regression models to check the regression output of each model with temperature sensitivity. The authors also noted that the energy savings in demand response programs relied heavily on the underlying baseline model that was chosen. Wang *et al.* [51] reviewed various data mining techniques for load profiling on the basis of different types of clustering. Zhang *et al.* [52] developed a demand response strategy for home HVACs by modeling their load consumption by integrating regression model with the neural network. The summary of all these schemes is given in Table 2.2.

Table 2.2: Data analytical techniques based on regression and classification.

S. No.	Scheme	Technique	Parameters	Objective
1.	Yamaguchi *et al.* [44]	k-means clustering, multiple linear regression models	Temperature	Studied the effect of outside air temperature on the regression models.
2.	Zhang *et al.* [45]	k-means clustering, decision tree	Consumption pattern, temperature	Develop correlations between normal and event days.
3.	Zhou *et al.* [46]	Least square, Lasso and ridge regression, k-NN regression	Hourly consumption, temperature	Understand consumer behavior in various demand response scenarios.
4.	Hatton *et al.* [47]	Control group selection, Regression	Consumption, accommodation type, area, devices, price	Devise a new control group selection policy on the basis of initial load curves.
5.	Woo *et al.* [48]	Regression	Day of week, month, weather, location, size	Modeled hourly regression to understand the effect of pricing.
6.	Klaassen *et al.* [49]	Multiple regression	Weather, daytime, day of week	Modeled the load profile of washing machines using regression.
7.	Jazaeri *et al.* [50]	Regression model, neural network	Temperature, day, load of past day, weekday, load of last week	Compared different models to check temperature sensitivity on regression results.
8.	Wang *et al.* [51]	Direct and indirect clustering	Voltage levels, temporal resolutions, distance metrics, number of clusters	Reviewed various data mining techniques for load profiling on the basis of different types of clustering.
9.	Zhang *et al.* [52]	Regression, neural network	Temperature	Developed a machine learning model for learning HVAC consumption pattern.

2.1.1 Data analytics for theft detection

As described above, data analytics has been used to cater to various research problems in smart grid. One such problem which is very important to handle, especially in developing nations, is the electricity theft. This problem can be solved in an efficient way using data analytics. As noted in a recent study by Northeast Group LLC, more than \$89.3 billion is lost every year worldwide due to the electricity theft [38]. India alone losses nearly \$16.2 billion per year, witnessing the highest levels of electrical theft. Electric theft basically refers to the intentional and illegal usage of electricity by various means. Tapping is one of the major causes of electrical theft and constitutes nearly 80% of the total losses [53]. Apart from this, tampering with the electric meters and billing alterations by the legitimate employees are other causes of electric theft [53,54]. Cybertampering is also one of the ways to maliciously alter the values of energy consumption profiles of meter data, which ultimately leads to reduced electricity bills [55]. The most commonly used malpractices for electricity theft have been depicted in Fig. 2.2.

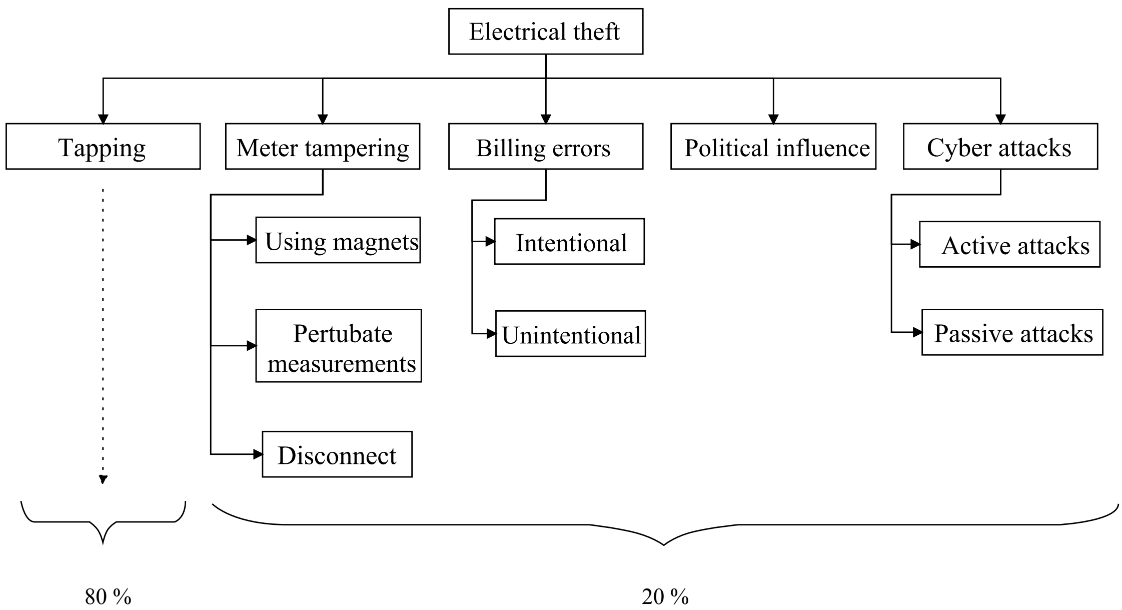


Figure 2.2: Different types of electricity theft.

Various researchers have also identified the importance of security for preventing theft in smart grids using ICT. For example, Guo *et al.* [55] proposed a three level framework to validate the credibility of data from home energy meters. The authors identified the fraudulent users based on the real-time measurements from feeder units and historic consumption pattern of the users. Giani *et al.* [56] proposed a system based on secure phasor measurement units (PMUs) for counterfeiting cyber attacks in smart grid. Though the concept proposed by the authors was novel, but it involved PMUs which are essentially used at transmission level to detect grid disturbances. Another technique which has been used for theft detection is artificial neural networks (ANNs). Muniz *et al.* [57] formed a dataset and trained the related model using ANN. Later on, fuzzy classification was used for further improvement, but this improved version also resulted in high false positives. False positives refer to test results where the legitimate customers are wrongly indicated as fraudulent. Costa *et al.* [58] proposed an ANN-based scheme to discover knowledge in databases for classifying the consumers as malicious. However, this model was found to be ineffective during uneven distribution of records. Moreover, the scheme witnessed low precision which eventually leads to large false positives.

In addition to the above mentioned approaches, researchers have extensively explored different types of techniques for theft detection. For instance, Angelos *et al.* [59] used fuzzy classification for identifying anomalies in usage pattern of customers. The major limitation of this proposal was that it failed to work in scenarios where fuzzification of features was a difficult task. Cabral *et al.* [60] used rough sets for identifying frauds in electricity usage but it produced high false positive rate. Another prominent data analytical technique which has been extensively used in the field of theft detection is SVM. Jokar *et al.* [39] used SVM for detecting various types of anomalies in the electricity consumption. However, the major drawback of this scheme was manual data collection from the customers which was an infeasible approach. Nagi *et al.* [61] also presented an SVM-based approach for detecting frauds based on the irregularities in the customers' consumption patterns. However, these irregularities need to be combined with other parameters for successfully identifying fraudulent customers with reduced false positives. Depuru *et al.* [62]

presented a classification model using SVM for labeling customers either as genuine or illegal. In another work [63], the authors utilized ANN with SVM to get better efficiency. Unfortunately, in both the above mentioned schemes, simple assumptions were chosen by the authors to identify fraudulent customers which could be easily violated. Apart from the above mentioned approaches, Nagi *et al.* [64] used fuzzy logic to improve the performance of SVM. The fuzzy inference system was used to process the list of customers derived from SVM-based fault detection scheme. However, the hit rate achieved using this approach was still low that suggested a high misclassification with respect to fraudulent consumers' detection.

From the above discussion, it can be summarized that the above mentioned approaches lag behind in the following ways.

- The false positive rate is high in terms of theft detection. This, in turn, increases the overall inspection cost.
- Most of the existing techniques have focused on identifying fraudulent consumptions at the consumers' end only. These approaches completely neglected the thefts from generation to distribution level.
- Majority of the parameters as discussed in the above approaches are found to be consumer centric in terms of electric usage.

2.2 Demand Response Management

The demand response in smart grid has been managed for various types of loads such as–residential, commercial and PHEVs. As far as residential loads are concerned, managing their demand response has many advantages like savings in their electricity bill and load reduction at the grid [65]. In this direction, many researchers have proposed different schemes to manage their demands. For example, Zhou *et al.* [66] used auction-based strategy to manage the demand at grid. The authors invited bids from the end users to shed their loads when supply was less and accepted the bids which maximized the grid benefits with minimum additional costs. The authors modeled demand response auctions on the basis of expressive power, truthful information, computational and economic efficiency. Working in the similar direction, Safdarian *et al.* [67] proposed a decentralized approach to manage the load profile of homes such that the overall load stability on the grid can be improved. The authors used mixed integer linear programming to solve the load optimization problem in smart homes. The drawbacks of this approach are that it may take time to converge to an optimal solution and consumers' load were modified every time when the grid's load profile was updated. Both of these conditions can hamper the smooth working of demand response management process. Apart from this, Lu [68] used heating, ventilation, and air-conditioning (HVAC) loads in homes and calculated the optimal number of HVAC units required to balance the load on the grid. The results obtained from this scheme depict that the HVAC loads provide an effective load balancing service in the smart grid and can become the potential source of revenue for the load management entities. Costanzo *et al.* [69] presented a load management scheme in smart buildings which was based on mixed integer linear programming (MILP). Authors proposed an admission control algorithm to control the operations of appliances so as to minimize the consumption cost with an emphasis on reducing the overall energy consumption in a home.

In other schemes, authors in [70] used Markov decision process to schedule the appliances in homes so as to decrease the power balancing cost in their neighborhood. The authors formulated a multi-stage stochastic optimization problem to model the working of deferrable appliances and applied approximation techniques to solve this problem. In [71], the authors used a greedy approach to manage the demand response in homes to decrease the price of electricity. For this purpose, the users identified the possible start times of the devices in dynamic pricing scenario and then a greedy iterative algorithm was employed for optimal device scheduling. Pipattanasomporn *et al.* [72] presented a home energy management system to manage the load of the homes according to their priority. The demand limit was assigned to the homes and whenever it was exceeded, the load was turned off based on their priority set by the user. Ozturk *et al.* [73] proposed a home energy management scheme to improve the demand response of the homes. This scheme was based on time of use, user constraints and pricing. Authors used branch and bound technique to schedule the devices intelligently based on the predicted consumer profiles. However, the simultaneous working of appliances in individual smart home in an uncoordinated manner may result in a new rebound peak [72]. Moreover, none of these schemes take the advantage of historical data gathered from smart devices in smart homes to achieve load stability.

Apart from the residential loads, PHEVs also have a huge impact on the electricity market because of their increased popularity in the transportation sector [74]. These PHEVs can be leveraged to manage the overall load on smart grid by regulating their charging requirements. Many studies exist in the literature which considered the load demand of PHEVs to manage their demand response. For example in [75], Fan proposed a distributed mechanism for handling the charging rates of PHEVs' batteries according to the available price and user's willingness to pay. The author modeled the user preferences as the criteria for user willingness to pay which indicated user's priority. Based on this priority, a charging scheme for PHEVs in smart grid was devised with which the users can adapt the charging rates for their PHEVs. Similarly, a price control mechanism for charging infrastructure to manage the charging requests of PHEVs based on an auto-regression model was provided in [76]. The charging requests were managed by regulating the electric supply for charging stations in such a way that the overall load stability on the grid was maintained while minimizing the waiting time for the customers. In [77], the authors proposed a peak load optimization strategy for PHEVs based on real-time prices to ease off the burden on grids during peak hours. This strategy was based on real-time regional pricing and wind power output data, and helped to shift peak in the load demand on the basis of different charging time schedule, charging speed and battery capacity of the vehicles. However, all these aforementioned studies focused on in-

Table 2.3: Summary of the demand response related schemes.

Schemes	**Technique used**	**Type of data**	**Application area**	**Objective of the scheme**
Zhou *et al.* [66]	Randomized auction	Smart grid	Demand response management	The bids from the end users were invited and accepted which maximized the grid benefits.
Safdarian *et al.* [67]	Mixed integer linear programming	Homes	Demand response management	The load optimization problem in individual smart homes was solved to manage load on the grid.
Lu [68]	Direct load control	Homes	Load balancing	The optimal number of HVAC units in homes was calculated required to balance the load on the grid.
Costanzo *et al.* [69]	Mixed integer linear programming	Buildings	Demand response management	An admission management scheme was proposed to control the operations of appliances for minimizing the cost.
Chang *et al.* [70]	Markov decision process	Homes	Home energy management	A decentralized algorithm was designed to locally compute the scheduling solutions for the users to decrease the cost.
Chavali *et al.* [71]	Greedy algorithm	Homes	Home energy management	A greedy approach was proposed to manage the demand response in homes to decrease the price of electricity.
Pipattanasomporn *et al.* [72]	Load shifting	Homes	Home energy management	An energy management system to manage the load of the homes according to their priority.
Ozturk *et al.* [73]	Branch and bound	Homes	Home energy management	Improved the demand response of the homes based on time of use, user constraints and pricing.
Fan [75]	Congestion pricing	PHEVs	Demand response management	The charging rates of PHEVs' batteries were modified according to the available price and user's willingness to pay.
Ban *et al.* [76]	Auto-regression model	PHEVs	Demand response management	Price control mechanism was developed for charging infrastructure to manage the charging requests of PHEVs.
Yao and Gao [77]	Stochastic modeling	PHEVs	Demand response management	Peak load optimization strategy to ease off the burden on grids during peak hours.

creasing the price during peak hours; but in reality, increased price may not always have a significant effect on the load demand of PHEV owners. It is because the PHEV owners might be willing to pay extra to get the charging facility. The summary of these schemes in terms of their techniques used, type of data, application area and objective of the scheme is given in Table 2.3.

The comparative analysis of few of the existing schemes is summarized in Table 2.4. This comparison is carried out on basis of various parameters such as–technique used, type of data considered, accuracy, error and type of application. Many of the existing techniques in the literature used SVM for forecasting the energy consumption in homes, which can be used in managing the demand response of smart homes. The authors in [78] have predicted the energy consumption of a single multi-family residential building based on SVM regression. The sensor-based hourly data of consumption and temperature were taken as the parameters to train the SVM. However, there are many other factors, which were not considered, that affect the energy usage in a home. Additionally, the scheme was not tested on other buildings to validate its effectiveness. In [79], authors collected data from three homes to forecast their electricity consumption. They compared multiple techniques of machine learning and found that least square (LS)-SVM worked well on the considered dataset. Thus, the data presented in Table 2.4 for [79] is of LS-SVM. The prediction error in this scheme was small, but only the consumption data of three homes was considered by the authors to validate their scheme. Chitsaz *et al.* [80] used self-recurrent wavelet neural network to forecast the load of buildings. Authors compared their scheme with simple wavelet neural network (WNN) and multi-layer perceptron (MLP), and noted that the error rate of their scheme was less than that of WNN and MLP. However, the authors used only the load data to train their neural network and their scheme also showed large forecasting errors on some days. Nayak *et al.* [81] used particle swarm optimization (PSO) technique to manage the demand of residential loads. This technique was able to reduce the peak in energy demand by 21%; but the authors used only the consumption data of various appliances by the users as parameters in this scheme.

Table 2.4: Comparative analysis of various demand response management schemes.

Schemes	**Technique used**	**Type of data**	**Accuracy**	**Error**	**Application area**
Jain *et al.* [78]	SVM	Homes	Medium	Low	Energy consumption forecasting
Edwards *et al.* [79]	LS-SVM	Homes	-	Low	Energy consumption forecasting
Chitsaz *et al.* [80]	Neural network	Building	Medium	Medium	Load forecasting
Nayak *et al.* [81]	PSO	Homes	Low	-	Demand side management
Shafie-khah *et al.* [82]	MILP	EVs	-	-	Demand response management
Soroudi *et al.* [83]	MINLP	Energy storage systems	-	-	Energy loss minimization.

Authors in [82] used EVs as aggregation agents and analyzed their participation in various demand response programs. The authors assumed fixed duration hours for peak and valley in parking lots and used Gaussian distribution to model the EVs behavior. Another scheme that emphasized on demand response management was presented in [83] in which the hourly active loss payment to the user by network operators was minimized by managing the optimal schedule of demand response and energy storage systems. The authors noted that demand response management effectively reduces the active loss payment; however, they did not mention that how the connected nodes was controlled.

Besides using the residential sector and PHEVs, many researchers have used commercial buildings to minimize their load demands so as to balance the load profile of the grid. For instance, the authors in [84] listed some basic actions that can be followed in a commercial building so as to improve the load consumption in such buildings. Zhou *et al.* [85] studied the effect of demand response for reducing the load in commercial buildings and proposed an agent-based framework for the same under different operational environments. The authors noted that this study was able to take such demand response actions which shaved the overall load profile of the grid during peak hours and also reduced the volatility of electricity demand. Chai *et al.* [86] minimized the energy consumption costs for scheduling the meetings in a commercial building. The problem of scheduling the meetings was modeled as a MILP problem and the authors presented a heuristic algorithm to solve this problem. The proposed solution provided a near optimal solution set while significantly reducing the computation time in comparison to the commercial solvers. Logenthiran *et al.* [87] managed the demand response of the loads connected to the smart grid with the help of an evolutionary algorithm which was based on shifting of loads to off-peak hours. In [88], authors ranked workload units in a manufacturing industry for vehicle cockpit assembly as potential candidates for reducing the overall load without interfering with the feasibility constraints of this industry. The authors used a fuzzy inference system to rank the workstations and used this ranking while formulating the workload scheduling problem as an integer programming problem. In another study [89], authors utilized the on-site generation to reduce the overall load on grid during peak hours for the same industry. Authors formulated the workload unit scheduling problem as a non-linear mixed-integer optimization problem by taking into account the factors related to operational, financial and energy constraints. To reduce the load in commercial buildings, the authors in [90] modeled the non-linear dynamics of HVAC systems according to the market mechanisms on the basis of multiple agents. The dynamics of every unit of HVAC system in the building were modeled in order to calculate the constant parameters of the system from the gathered data so as to reduce their energy consumption. Apart from this, authors in [91] predicted the energy consumption of the air circulation fan in a commercial HVAC system for reducing the overall load. The comparative analysis of the aforementioned schemes is carried out in terms of the method used, building type, nature of scheme, consumer comfort, energy savings and objective as given in Table 2.5.

Table 2.5: Comparative analysis of the commercial schemes for load stabilization.

S. No.	Scheme	Method used	Building type	Nature of scheme	Consumer comfort	Energy savings	Objective of the scheme
1.	[84]	Basic actions	University building	Distributed	×	Low	Implemented several basic strategies to improve energy usage efficiency in the building.
2.	[85]	Agent-based modeling	Commercial buildings	Distributed	×	Low	Explored the effects of various demand response models on different market scenarios.
3.	[86]	MILP	Commercial building	Distributed	✓	Medium	Scheduled the meetings using heuristic approach to save electricity.
4.	[87]	Evolutionary algorithm	Residential, commercial, industrial	Centralized	×	Medium	Minimized the difference between final load curve and objective load curve.
5.	[88]	Fuzzy system	Manufacturing industry	Distributed	×	Low	Studied the participation of workload units in demand response event while satisfying their operational constraints.
6.	[89]	Non-linear mixed-integer optimization	Manufacturing industry	Distributed	×	Low	Optimized the working of assembly units for energy consumption by considering various operational and technical constraints.
7.	[90]	Transitive control approach	Commercial building	Distributed	✓	Low	Load reduction at the grid by reducing the energy consumption of building's HVAC systems.
8.	[91]	Single state variable scheme	Commercial building	Distributed	✓	-	Predicted the energy consumption for circulation fan of HVAC system to minimize its energy consumption.

2.3 Data Analytics-based Demand Response

Sakurama and Miura [92] showed that the demand response systems can become very effective when the energy consumption information of consumers is available. There are many data analytical based schemes reported in the literature which have been widely used by the researchers to manage the demand response in homes and buildings. The underlying principle behind these schemes is to analyze the gathered data to provide the demand response solutions. The analysis of data can be carried out using various techniques such as–regression, reinforcement learning, SVM based and reward based mechanisms. Few of such schemes are discussed as follows. Huang *et al.* [93] proposed a demand response management scheme for smart buildings on the basis of a multi-agent minority-game to reduce the peak energy demand from the grid. The game was played between various smart homes to maximize the utility in the building after analyzing the gathered data from the homes. The main aim of this scheme was to reduce the power drawn from electric grid and to increase the utilization of solar power to manage the load demand. However, this scheme would not work in the absence of PV panels in the homes. In order to overcome this drawback, many other studies also exist in the literature which focused on the demand response management without utilization of PV panels [94, 95]. The authors in [94] modeled the day-ahead demand of residential heat pumps using reinforcement learning by exploiting the weather data. To model their schedule, the authors employed a model-free Monte Carlo method by considering the state-action value function. In [95], the load demand of electric water heaters was managed in residential homes. For this purpose, electric energy was converted to heat energy in the off-peak hours which can be used to cater the load demand during the peak hours. However, a very limited set of appliances were utilized in [94, 95] to manage the demand response. Moreover, in [95], the losses in conversion and storage of electric energy in form of heat energy were substantial. Vivekananthan *et al.* [96] managed the demand response of residential appliances to control the voltage at the

Table 2.6: Summary of the data analytics-based demand response schemes.

Scheme	**Technique used**	**Type of data**	**Nature of scheme**	**Objective of scheme**
[93]	Multi-agent game	Smart buildings	Distributed	To reduce peak demand on grid and fair utilization of solar power
[94]	Batch RL	Smart homes	Centralized	To minimize the difference between day-ahead plan and actual consumption.
[95]	Load shifting	Smart homes	Centralized	To provide balancing reserves for the utility in presence of wind generation.
[97]	MINLP	Smart homes	Distributed	To reduce the electricity bill and minimizing the amount of curtailed energy.
[96]	Reward mechanism	Smart homes	Distributed	To improve the voltage at feeder level.

feeder and prevent transformer overload. Authors computed various indexes from consumer survey and triggered demand response control events on their basis. For this purpose, the authors utilized the data gathered from different homes to reduce the network peaks. However, the authors used consumer survey to generalize the requirements of all the users; but in reality, every household has its own set of needs and priorities. Moreover, the authors did not discuss that how the demand response would actually be managed in real-time. The summary of these data analytical schemes for demand response management is given in Table 2.6. Moreover, the comparative analysis of these in terms of consumer benefits, consumer satisfaction, number of appliances considered, cost savings in homes and load reduction at grid is carried out as depicted in Table 2.7.

Table 2.7: Comparative analysis of the data analytical demand response schemes.

Scheme	**Consumer benefits**	**Consumer satisfaction**	**Number of appliances**	**Cost savings**	**Load reduction**
[93]	×	×	-	High	High
[94]	×	×	Two	Fair	Low
[95]	✓	-	One	High	High
[97]	-	✓	Limited	Fair	Fair
[96]	✓	✓	Limited	Fair	Fair

2.4 Home Energy Management Systems

Apart from the demand response schemes discussed above, the recent trend is focused on developing the home energy management systems (HEMS) to manage the load demand in homes as the residential sector is the biggest electricity consumer out of all the electricity consumers [1]. Therefore, managing the energy supply in the homes can help to stabilize the load demand in the smart grid. Fig. 2.3 shows the energy consumption of various appliances in a typical home of U.S. [2]. It is necessary to understand the demand of these appliances in order to manage their load requirements in the smart homes. These loads should be managed in such a way that the overall burden on the grid is reduced without compromising the user comfort in smart homes. For this purpose, the energy management systems in homes can be deployed in two fashions, i.e., distributed and centralized. In distributed, each HEMS is responsible for the energy consumption in its smart home, while in centralized HEMS, it is responsible for the energy consumption in multiple homes. Many authors proposed various home energy management systems in this regard which are discussed as follows and the summary of which is given in Table 2.8.

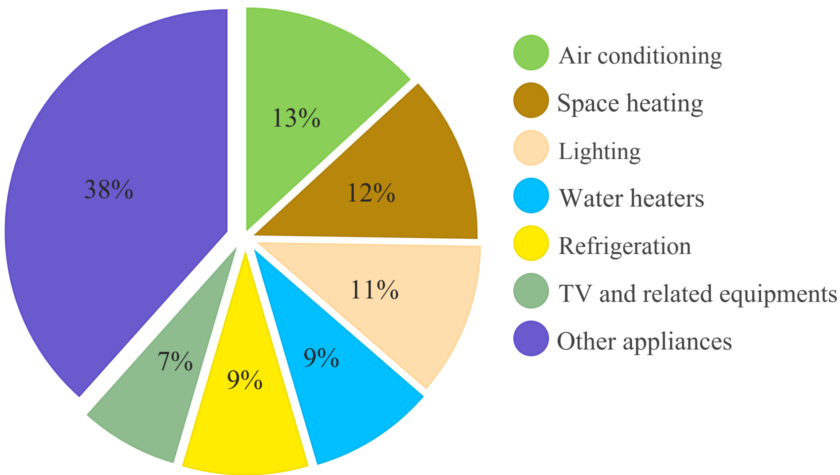


Figure 2.3: Energy usage in the typical U.S. homes [2].

In centralized energy management systems, Tushar *et al.* [98] proposed a Stackelberg game to control the joint ownership of energy between various residential units and the storage facility controller. The authors used the auction pricing concept to strike a deal between these entities for deciding the price at which the energy would be sold in the market. In another scheme, Tushar *et al.* [99] used of RES for the shared energy storage controller so as to minimize its operational cost. The authors introduced the concept of virtual cost for this purpose and used the theory of maxima and minima for optimizing this cost. The authors also designed an algorithm to take energy trading decisions for selling the energy to other entities after managing the energy demand at the shared controller facility. Byun *et al.* [100] presented a cloud-based home energy management system to tackle the issues of intermittent nature of RES. This system assigned a real-time priority to a device in the home according to its type and present working status. The RES were then used to power the devices in accordance with their assigned priority.

Zhao *et al.* [101] proposed a system for efficient appliance scheduling in smart home so as to reduce the peak to average demand in smart grid. The authors combined real-time pricing and inclining block model for devising a power scheduling scheme which effectively reduced both the electricity cost and peak to average ratio. Wang *et al.* [102] proposed a strategy of shared energy storage at consumer end for reducing the overall energy prices in the network and to support low voltage distribution networks when required. The main aim of this study was to find an optimal trade-off between energy prices and demand response in residential homes so as to reduce the burden on the distribution network as well as increase the benefits of the consumers and network. Shakeri *et al.* [103] proposed an energy management system which uses battery energy storage systems as well as managed the temperature of thermal appliances to ensure that the power consumption in a home was less than a certain set level. The proposed system used the price information sent by the utility company to store the energy in batteries in off-peak hours and utilized this energy to manage the load during peak hours.

In the distributed energy management system category, Chen *et al.* [104] considered the intermittent power generated from RES and the uncertainties involved in operating the devices while scheduling the devices in homes for the purpose of cost minimization. The initial schedule was prepared using linear programming while stochastic scheduling was used to handle the uncertainties. Paterakis *et al.* [105] developed an energy management system for homes in which the day ahead schedule of appliances was prepared on the basis of available pricing information and characteristics of each appliance. The authors modeled various appliances in the homes by using MILP technique with an aim to minimize the cost of electricity in the homes. Jo *et al.* [106] presented an energy management system for managing the loads of heating and air conditioning devices to minimize the energy consumption in smart homes as well as to enhance the consumer comfort. For this purpose, the real-time prices were used in branch and bound technique for creating an efficient schedule of these devices.

In addition to these schemes, Chen *et al.* [107] presented an energy management system for managing the energy demand in buildings such that the aggregated load demand of these building was matched with the desired load demand so as to maintain a balance between demand and supply. To find an optimal load demand of a region, the authors used the average consensus algorithm for distributing the energy to each building energy management system. Han *et al.* [108] designed a power line communication based distributed solution for controlling the energy consumption in smart homes. In this scheme, the local controller in homes estimated the energy generation of RES based on weather forecast. Using this information, the controller manages the usage of energy in such a way that it minimizes the total energy consumption cost in smart home. Anvari *et al.* [109] developed a multi-objective MINLP-based smart energy management system for home users to reduce their energy consumption while respecting the user comfort. Authors proposed a system to reduce the energy usage in the homes to ensure that the thermal zone for the inhabitants remains within the comfortable limit.

Table 2.8: Comparative analysis of various home energy management systems.

Schemes	**Technique used**	**Use of DERs**	**Nature**	**Type of load**	**Objective of scheme**
[98]	Game theory	Yes	Centralized	Residential community	Auction pricing between shared energy resources and residential units for joint ownership of energy.
[99]	Theory of maxima and minima	Yes	Centralized	Residential community	Minimize the operational cost of shared energy storage facilities using RES.
[100]	Dynamic priority	Yes	Centralized	Smart homes	Tackle the issue of intermittent nature of RES to power loads in smart homes.
[101]	Inclining block model	No	Centralized	Smart homes	Efficient appliance scheduling scheme for reducing the peak to average ratio.
[102]	Shared energy storage	Yes	Centralized	Residential units	Energy storage at consumer end for reducing the energy prices and to support low voltage distribution networks.
[103]	Energy storage	Yes	Centralized	Smart homes	Optimized the energy consumption in homes based on price when it is more than a threshold limit.
[104]	Linear programming	Yes	Distributed	Smart homes	Device scheduling based on intermittent power generated from RES and the uncertainties involved in the system.
[105]	MILP	Yes	Distributed	Smart homes	Optimal appliance scheduling for shaping the load profile on the basis of day ahead pricing information.
[106]	Branch and bound	No	Distributed	Smart homes	Energy management system for managing heating and air-conditioning devices for enhancing consumer comfort.
[107]	MILP	No	Distributed	Buildings	Match the overall load demand of a region with desired demand profile.
[108]	Power line communication	Yes	Distributed	Smart homes	Minimize the total energy cost in homes on the basis of power generated by RES.
[109]	MINLP	Yes	Distributed	Smart home	Minimize the energy use while maintaining the comfort lifestyle of the users.

2.5 Applications Areas in Smart Grid

There are many other application areas which complement the demand response management and data analytics in smart grid. One such research area that has been explored in this thesis is the dimensionality reduction of the data gathered in the smart grid environment. Munshi and Mohamed [110] noted that the data gathered in smart grids would rise to an extent and it would be infeasible to apply the present frameworks to process this huge amount of data. In this regard, Zhou *et al.* [111] discussed various characteristics of energy related data from the aspects of power generation, asset management, renewable energy management and demand side management. However, the enormous size of gathered data requires the adoption of advanced data analytical techniques so as to process it efficiently [112].

To cater this issue, many proposals exist in the literature which focus on data dimensionality reduction for efficient processing of data. For instance, Silipo *et al.* [113] analyzed seven such techniques to reduce the dimension in the data namely missing values, low variance filter, high correlation filter, principal component analysis, random forests, feature elimination and feature construction. The authors noted that feature elimination method gave best reduction rate while having maximum accuracy on the validation set. However, they tested these approaches on a fairly small number of data entries. Moreover, as already discussed, these techniques would perform poorly on heterogeneous data and are not scalable to very large datasets [114]. To overcome these challenges, a tensor-based approach can be utilized to reduce the dimensionality of data gathered in the smart grid environment.

Many studies exist in the literature that used tensor-based data representation and dimensionality reduction for various application domains. For example, Zhang *et al.* [115] proposed a dimension-reduction scheme for feature extraction of hyperspectral images. To represent a pixel's spectral-spatial feature information, a tensor organization scheme was developed and optimized on the basis of tensor discriminative locality alignment by removing redundant information. Sun *et al.* [116] proposed multi-variant incremental tensor frameworks to compute the summary of higher-order data and to find the hidden correlations between the data. These variants are namely dynamic tensor analysis, streaming tensor analysis and window-based tensor analysis which were used for summarizing large tensors, thereby saving space and detecting patterns. Zhu *et al.* [117] proposed a predictive model to deal with tensor inputs while also considering the domain knowledge for wafer quality prediction gathered from various sources. The authors assigned weight tensors by approximating the low-rank tensors and then estimating the low-rank approximation using the gathered domain knowledge. Kuang *et al.* [118] presented a tensor-based scheme for dimensionality reduction to store the heterogeneous data (unstructured, semi-structured and structured). The authors represented the data in terms of subtensors which were merged to form a unified tensor model. The core tensor containing useful information was extracted using higher order SVD which updated the orthogonal bases and computed new core tensors. All these aforementioned schemes indicate that tensor-based technique is quite useful in reducing the dimensionality of data. The comparative analysis of the existing schemes in terms of data dimensionality is given in Table 2.9. This comparison is carried out in terms of technique used, type of data, nature of scheme, reduction percentage, reconstruction error and application area.

Table 2.9: Comparative analysis with respect to data dimensionality.

Scheme	**Technique used**	**Type of data**	**Nature of scheme**	**Dimension reduction**	**Reduction %age**	**Reconstruction error**	**Application area**
[113]	Multiple	Structured data	Centralized	✓	Medium to high	-	Dimensionality reduction.
[115]	Tensor discriminative locality alignment	Hyperspectral images	Centralized	×	-	-	Feature extraction from images.
[116]	Incremental tensor analysis	Structured data	-	✓	-	20%	Anomaly detection.
[117]	Block co-ordinate descent	Semiconductor manufacturing data	-	✓	-	-	Wafer quality prediction.
[118]	Incremental HOSVD	Video and XML data	-	✓	High	Low (7%)	Dimensionality reduction on big data.

2.6 Research Gaps

Although a number of research proposals exist in the literature which addressed various problems in demand response management, but still these proposals lack in many aspects. There still persists the need for a new solution which fulfills the number of research gaps that exist today. Based upon the review of the literature, following research gaps have been identified in the existing solutions:

- An optimized demand response management scheme is lacking in the literature which considers various factors such as power generation from intermittent sources like solar PV panels and distributed energy sources (DERs), and also takes consumer's behavior into account while providing demand response.

- The dynamic changes in the load requirements need to be handled in the near real-time so as to maintain the load stability in smart grid. The use of DERs to maintain the grid stability in such a scenario can prove fruitful. But a unified strategy is lacking which considers the uncertain nature of DERs to maintain the load stability of the grid.

- The integration of data (being generated in smart grid environment) with grid operations to enhance the efficiency of the system is lacking.

- The technological shift has seen the urge for utilities to move to data analytics to stay competitive in business but the lack of customers interest (due to security and privacy) and uncertainty involved in the complete system proves to be a barrier in generating the optimal and workable business models.

- The false positive rate in predictive analytics such as fraudulent consumers in theft detection is very high in existing approaches which in turn increases the overall inspection cost.

- The complete solutions for data analytics are yet to be realized in smart grid environment due to lack of techniques and frameworks which guarantee the quality of service under different application behaviors due to their uncertain and unpredictable nature.

- The heterogeneity in data gathered from multiple sources and real-time streaming of data are yet another issues in performing data analytics on data generated in smart grid environment.

- The big data and cloud computing issues also need to be taken care of in addition to the challenges faced in smart grid environment for proper analytics of data. These challenges put an additional burden on data analytics and need to be properly handled.

2.7 Objectives

By analyzing the research gaps found in related work, following objectives are set-forth:

1. To perform data analytics on the data generated by various devices in Smart Grid.

2. To design an efficient scheme for managing demand response in Smart Grid environment.

3. To test and validate the proposed schemes in Smart Grid environment. Their comparison with the existing techniques with respect to various evaluation metrics would also be done in order to find out the efficient strategy for demand response.

Chapter 3

Data Analytics for Theft Detection

Electric theft is one of the most prominent issues pertaining to the conventional power grids and has been a major concern to the utility providers for quite a long time. According to a recent survey conducted by Northeast Group, LLC, more than $89.3 billion are lost every year worldwide due to the electricity theft [38]. Keeping this in mind, a comprehensive top-down scheme based on decision tree (DT) and support vector machine (SVM) is presented in this chapter to detect electricity theft in complex power networks[1]. Unlike existing schemes, the proposed scheme is capable enough to precisely detect and locate real-time electricity theft at every level in power transmission and distribution (T&D). The proposed scheme is based on the combination of DT and SVM classifiers for rigorous analysis of gathered electricity consumption data. In other words, the proposed scheme can be viewed as a two level data processing and analysis approach, since the data processed by DT is fed as an input to the SVM classifier. Furthermore, the obtained results indicate that the proposed scheme reduces false positives to a great extent and is practical enough to be implemented in real-time scenarios.

[1]The contents of this chapter are published in: A. Jindal, A. Dua, K. Kaur, M. Singh, N. Kumar, and S. Mishra, "Decision Tree and SVM-based Data Analytics for Theft Detection in Smart Grid," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 3, pp. 1005-1016, 2016.

3.1 Contributions

The major contributions of this chapter are as follows.

1. The primary contribution of this chapter is to present a comprehensive top-down approach to detect the electric theft in smart grid environment. This implies that the proposed approach detects the electrical theft at different levels, i.e., transmission, distribution and consumer levels.

2. Another significant contribution of this chapter is the combination of SVM classifier with DT-based approach for reducing false positives to a great extent.

3. Furthermore, the proposed scheme has taken diverse set of parameters for classification purposes such that it closely relates to the real world scenarios.

3.2 Proposed Scheme

This section illustrates the working of the proposed scheme along with the architectural diagram to gather the data in smart grid as shown in Fig. 3.1. As clearly depicted in this figure, smart grid comprises of numerous interconnected entities which include power generation, transmission lines, distribution stations and end users. Initially, the power is generated at power generation unit and is transmitted to the different distribution stations through high-voltage transmission lines. After this, the power is redistributed across various sectors and areas. Usually, electrical theft may occur at any level of power transmission as mentioned above. In order to identify these thefts, the proposed scheme employs data analytics on data aggregated from various entities. For collecting the related electricity consumption data on a real-time basis, smart meters and sensors are assumed to be deployed across all levels. The data from smart meters and sensors is gathered through the Internet. Smart meter uses wireless medium to communicate the readings of the meter to the utility. Once this data is collected, it is then relayed to a central utility server for

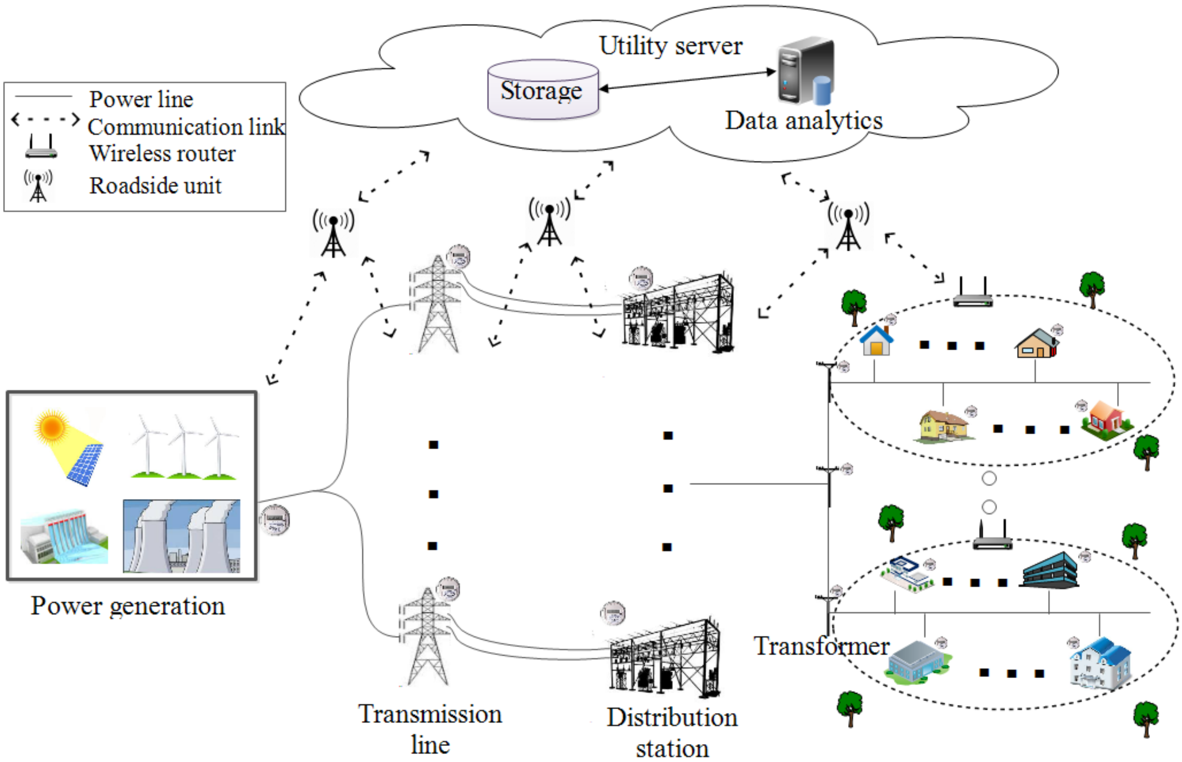


Figure 3.1: Data aggregation and processing in smart grid environment

further processing using ICT [119, 120]. This has been depicted in Fig. 3.1, where the server performs the analytics on the received data using SVM and DT to precisely identify the potential areas of the electric theft in the distribution network.

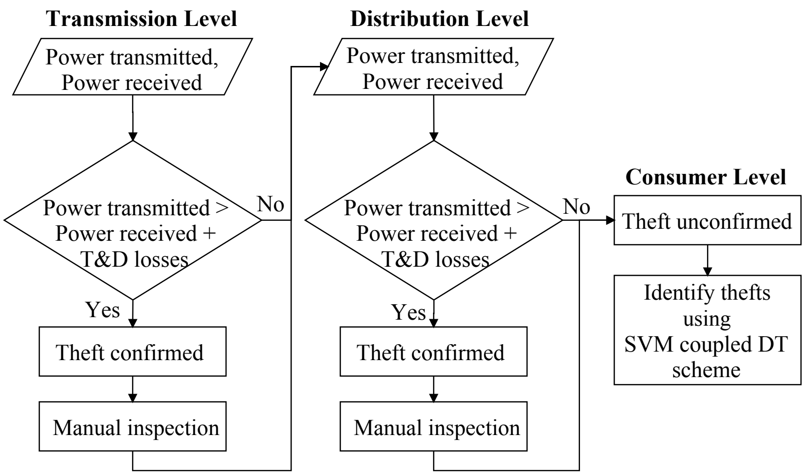


Figure 3.2: Flowchart of the proposed theft detection scheme.

The detailed working of the proposed scheme is depicted in the form of a flowchart in Fig. 3.2. Initially, the utility server compares the total power transmitted and the total power received at various levels, i.e., transmission and distribution. These computations are done on a real-time basis and in a seamless manner by the server. The utility server decides whether theft has occurred at a particular level or not on the basis of these computations. If the computed value of total power transmitted exceeds the power received (including T&D losses), then the server considers it as a theft at the corresponding level which is followed by the manual inspection. Otherwise, no theft is assumed to be occurring at the respective level and the same process propagates to the next level. The T&D (i.e., transmission and distribution) losses are computed using the following equation.

$$L_2 = (1 - \alpha)L_1 \tag{3.1}$$

where, L_1 is the level from where the power is transmitted, L_2 is the level where the power is received and α is the loss factor that varies with the load.

Apart from identifying the theft at T&D level, it is also vital to identify the thefts at the end users level which can occur in two ways. One being the physical tampering, where the consumer makes an unauthorized connection with the grid to steal electricity. Other being the electronic tampering of metered data to intentionally lower the energy consumption values. Both these types of theft will lower the energy consumption profile of end users; and the users participating in any of these are considered as fraudulent consumers. For handling these issues, this work presents a classification scheme to identify the fraudulent consumers involved in both types of electrical theft. Thereby, making the proposed scheme a top-down approach which finds theft at every level in the electrical network. The proposed classification scheme at consumer level combines two widely used classification schemes namely DT and SVM. Basic block diagram of this combination is depicted in the Fig. 3.3.

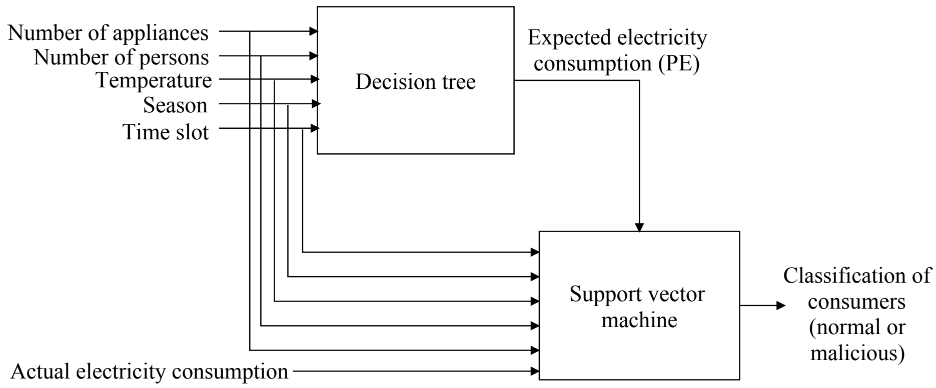


Figure 3.3: Combination of DT with SVM

It incorporates diverse set of parameters for classification of consumers as normal or malicious, as mentioned in Fig. 3.3. All these parameters are given as input to both the DT and SVM classifiers one by one. The entire process of this classification scheme can be understood in two steps. During the first step, DT classifier is employed to operate on the said parameters and it computes the expected electricity consumption of the consumers. Once this parameter is computed, then SVM classifier is made to operate on the basis of previously defined parameters along with this newly computed parameter. As a result of these computations, SVM finally detects the malicious consumers involved in electrical theft. Thus, the scheme attains higher relevance due to the combination of these two classifiers to yield greater precision. The detailed description about these classifiers are mentioned in subsequent sections respectively.

3.3 Decision Tree

DT refers to a classification technique that segregates the attributes into classes based on their respective features. During DT classification, the training set is classified into predefined classes and a discrete label is assigned to each entry of a class. This classification can either be done by the experts based on their experience, or by using different logical models. However, the classification performed by expert(s) is done at the expense of time and cost. The major advantage of DT is that it classifies the data using logical models with greater accuracy while overcoming the drawbacks of the former method. Therefore, the DT-based approach is used in the proposed scheme to precisely detect the malicious consumers.

In the proposed scheme, DT is used in theft detection by calculating the expected electricity usage (PE) of the consumers. This usage is given as an input to the SVM classifier which classifies the consumers as normal or malicious. For this, C4.5 algorithm [121, 122] based approach which is an extended version of ID3 algorithm [123] is used to construct the DT. In this approach, an attribute is selected to partition the samples based on entropy and information gain. Let S be set of samples and $F_r(C_i, S)$ be the frequency of samples in S that belong to class C_i. Assuming that there are k distinct classes and total number of samples in S are $|S|$. Then, the entropy (E) of S is calculated as follows [124].

$$E(S) = -\sum_{i=1}^{k}\left(\frac{F_r(C_i, S)}{|S|}.\log_2\left(\frac{F_r(C_i, S)}{|S|}\right)\right) \tag{3.2}$$

Once $E(S)$ is calculated, S is partitioned into n number of outcomes with respect to an attribute say x. Thus, the $E(S)$ with respect to x (i.e., $E_x(S)$) becomes the weighted sum of all the individual entropies of its subsets (S_i). The final entropy and information gain (IG) after the partitioning is calculated as follows [124].

$$E_x(S) = -\sum_{i=1}^{n}\left(\frac{|S_i|}{|S|}E(S_i)\right) \tag{3.3}$$

$$IG(x) = E(S) - E_x(S) \tag{3.4}$$

The IG is calculated for all the attributes and an attribute with maximum IG is selected to partition S which becomes the parent node of the decision tree. This procedure is repeated and the child nodes are created in a similar manner until all the entries are classified to a single output class.

For illustration, the above procedure is applied to the sample database depicted

Table 3.1: Sample data considered for decision tree

Att1	1	1	1	1	1	2	2	2	2	3	3	3	3	3
Att2	20	40	35	45	20	40	28	15	25	30	20	30	30	46
Att3	Y	Y	N	N	N	Y	N	Y	N	Y	Y	N	N	N
Output	1	0	0	0	1	1	1	1	1	0	0	1	1	1

in Table 3.1. The initial entropy before splitting the partition is computed as follows.

$$E(S) = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14} = 0.940 \tag{3.5}$$

The splitting can be done based on all the attributes present in the training dataset. If the splitting is done with respect to first attribute (i.e., *Att* 1) which comprises of three different values (1, 2 and 3), then $E_1(S)$ is calculated as shown below.

$$E_1(S) = \left.\begin{array}{l} \frac{5}{14}(-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}) \\ +\frac{4}{14}(-\frac{4}{4}\log_2\frac{4}{4} - \frac{0}{4}\log_2\frac{0}{4}) \\ +\frac{5}{14}(-\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}) \end{array}\right\} = 0.694 \tag{3.6}$$

The IG for this attribute is calculated using Eq. (3.4) as.

$$IG(1) = 0.940 - 0.694 = 0.246 \tag{3.7}$$

So, the IG by splitting the tree on basis of *Att* 1 comes out to be 0.246. The tree can also be split on the basis of *Att* 2 and *Att* 3 for which $IG(2)$ and $IG(3)$ comes out to be 0.103 and 0.048 respectively. It is evident that IG is maximum for *Att* 1. Thus, the DT is split on the basis of this attribute. The DT formed after splitting using *Att* 1 is depicted in Fig. 3.4. The leftmost and rightmost subtree are again split based upon the attribute that have maximum IG. This process is repeated for all attributes. The final DT after recursive splitting is depicted in Fig. 3.5.

The aim of proposed DT is to predict the electricity consumption of consumers. This is achieved by extracting the priorities of the features from the constructed

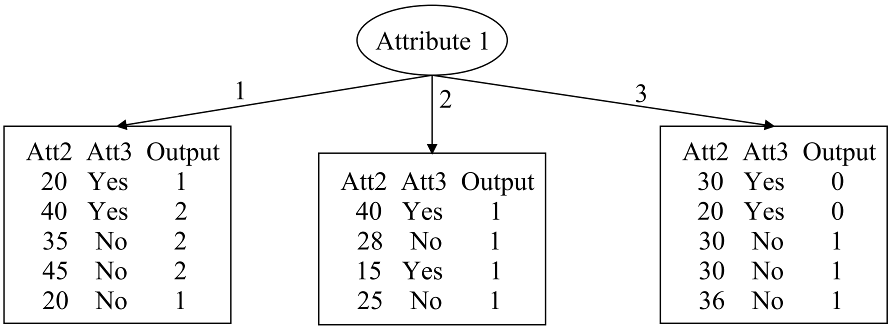


Figure 3.4: Decision tree after first splitting.

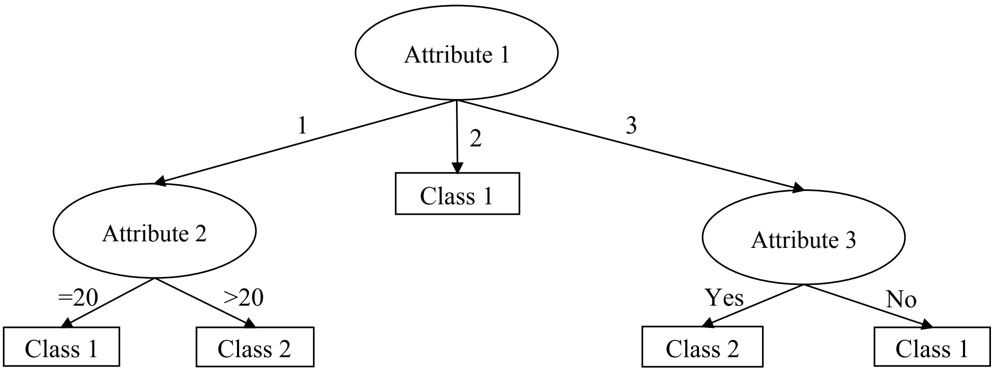


Figure 3.5: Final decision tree.

DT. For this purpose, the data is collected, relevant features are extracted and DT is formed using algorithm 1. The initial entropy, the entropy for all the features and IG are calculated using Eq. (3.2), (3.3) and (3.4) respectively. The feature having maximum IG after splitting, is selected as the parent node. This process is repeated until all the attributes are classified as child nodes and optimized DT is obtained.

After the formation of DT, weights are assigned to all attributes for predicting the expected electricity usage of a consumer. This prediction is done based on the priorities of the respective attributes. The priorities are decided according to the occurrence of an attribute in DT using algorithm 2. This algorithm results in the prioritization of attributes, using which the prediction of electricity consumption is done.

Once the priorities are set, the value of predicted electricity (PE) is calculated. For every attribute i, weight w_i is assigned to it based upon its priority. The contribution of specific attribute to PE is calculated using the weights assigned to

Algorithm 1 Formation of decision tree

Input: Features
Output: Decision tree (DT)

Collect the data
Extract the features
while ($N_{at} \geq 2$) **do** ▷ N_{at} = Number of features
    Calculate total entropy (E) using Eq. 3.2
    for ($i = 1; i \leq N_{at}, i++$) **do**
        Calculate E_i using Eq. (3.3)
        Calculate IG_i using Eq. (3.4)
        Initialize $Max = 0$
        if ($IG_i \geq Max$) **then**
            Max = IG_i
            k = i
        end if
    end for
    Make k as parent node
    Split the attributes on decision attribute k
    $N_{at} = N_{at} - 1$
end while

the respective feature. This weight is calculated for all the attributes and finally, PE returns the expected electricity consumption of the consumer using equation mentioned below.

$$PE = \sum_{i=1}^{n} w_i.V_i \tag{3.8}$$

where, V_i is the value pertaining to attribute i. The PE is then provided as input to SVM classifier which detects the malicious consumers using PE and other parameters. The detailed working of SVM classifier is explained in next section.

Algorithm 2 Setting the priorities

Input: DT

Output: Priority of features

```
n = number of features
Initialize i=1
function T_r(k)
    if (i == 1) then                                  ▷ Root node
        S(i,1) = k
        S(i,2) = 1                                    ▷ Top Priority
        i = i + 1
        T_r(left(k))
        T_r(right(k))
    else                                              ▷ Non-root node
        if ((left(k) ≠ NULL) || (right(k) ≠ NULL)) then
            for (a = 1; a ≤ i; a++) do
                if (S(a,1) == k) then
                    return S
                else
                    S(i,1) = k
                    S(i,2) = i                        ▷ Set Priority
                end if
            end for
        end if
        if (S(i − 1) == n) then                       ▷ All attributes explored
            return S
        else
            T_r(left(k))
            T_r(right(k))
        end if
    end if
end function
```

3.4 Support Vector Machine

SVM is the supervised machine learning method which is used for classification of data. The primary goal of using SVM is to classify the unseen data accurately by minimizing the classification error using a decision function. This is achieved by training the SVM on the training data and afterwards using it to predict the output class of the unseen data. In this section, SVM-based approach has been put forth for the purpose of theft detection. The parameters used in SVM classification are number of persons, number of appliances, temperature, time of day, season, actual consumption and predicted electricity consumption from DT. The various steps used in SVM classifier are as shown in Fig. 3.6 and are explained briefly as follows.

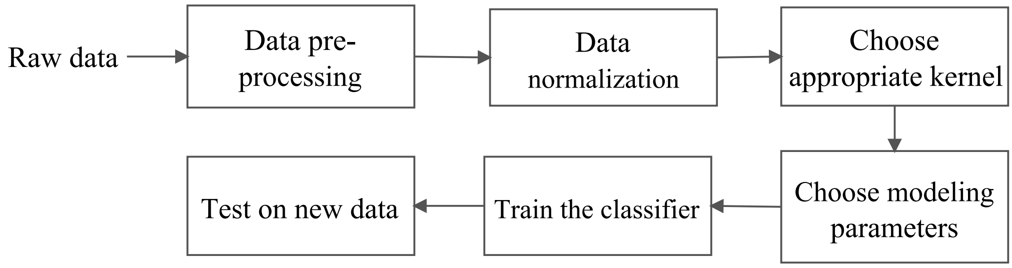


Figure 3.6: SVM classification process.

3.4.1 Data pre-processing

In this step, raw data from various sensors and smart meters deployed in the smart grid environment is collected and transformed into the format recognized by SVM classifier. For this, all the categorical attributes in the collected data need to be converted into numeric format.

3.4.2 Data normalization

Data normalization is one of the main strategies that needs to be employed before performing classification using SVM. The basic purpose of data normalization is to keep check on the attributes having greater range of numeric values, so that the smaller sized values are not neglected.

Let X be the data value such that $X \in D$, where D is the domain of X, then, the normalization of X can be performed using a normalization function as mentioned below.

$$X' = a + \frac{(X - X_{min})(b - a)}{X_{max} - X_{min}} \tag{3.9}$$

where, X' represents the data value after normalization. X_{max} and X_{min} are the maximum and minimum data values in D respectively. Apart from this, a and b

are the minimum and maximum values in the specified range respectively. In the proposed scheme, all the data values are scaled down in the range [-1,+1] using Eq. (3.9).

3.4.3 Choice of appropriate kernels

Choosing kernels according to the problem is also a very important task in SVM classification. One must choose the best available kernel that fits the respective problem. Different kernel functions can be used for different cases depending upon the type of data. Linear kernels are used in the cases where training set is easily separable or the feature set is large in number. For rest of the cases, non-linear kernels are used. These kernel functions are explained further as follows.

1. Linear kernels: The working of linear kernel is as follows. Let X is the training set such that,

$$X = \{x_1, x_2, \ldots x_n \,|x_i \in \Re^m\} \tag{3.10}$$

where x_i is the i^{th} training set which ranges from $i = 1, 2, \ldots, n$. The feature space of m dimensionality is represented by $\Re^m$. Its output would be in one of the two classes defined below.

$$Y = \{y_i \in (-1, 1)\} \tag{3.11}$$

This implies that the output would either belong to the class having output +1 or -1. Apart from this, the mapping (M) of the input training value and the output class can be represented as,

$$M = (X, Y) \tag{3.12}$$

The objective of linear kernel is to create a decision boundary (or hyperplane) which divides the initial training set into two linearly separable parts. This decision boundary should be at a maximum distance from the values of training set so that the test values can be easily classified into the two planes. The mathematical representation of this decision boundary is given as follows.

$$f(x) = w.x + b \tag{3.13}$$

where w is the normal vector to the plane and b represents the regularization parameter. The above function is primarily dependent on the value of $||w||$ which is the norm of vector w. For simplification, $||w||$ can be replaced with $\frac{1}{2}||w||^2$. This leads to the quadratic equation which can be represented as below.

$$min\ \frac{1}{2}||w||^2 \tag{3.14}$$

$$subject\ to$$

$$y_i(w.x_i + b) \geq 1 \tag{3.15}$$

To counter the occurrence of inequalities, the slack variables (ξ_i) are included in the classification problem as highlighted in the below equation.

$$y_i(w.x_i + b) \geq 1 - \xi_i, 1 \leq i \leq n \tag{3.16}$$

The optimization problem can now be represented as,

$$min\ \left(\frac{1}{2}||w||^2 + C\sum_{i=1}^{n}\xi_i\right) \tag{3.17}$$

where C is used to trade-off between maximum margin and minimum error. Eq. (3.17) is solved with the help of Lagrange multipliers (α_i,$\beta_i \geq 0$). Substituting these variables in the above equation, the final objective function can be represented as below.

$$max \left(\sum_{i=1}^{n} \alpha_i - \frac{1}{2} \alpha_i \alpha_j u_i y_j (x_i, x_j) \right) \tag{3.18}$$

$$subject\ to$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0, \alpha_i \in [0, C] \tag{3.19}$$

The decision boundary equation will be consequently be written as

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i (x_i, x_j) + b \tag{3.20}$$

2. Gaussian RBF kernels: Gaussian radial basis function (RBF) kernel is the most commonly used non-linear kernel in software packages that separates the solution sets which are not linearly separable [125]. This kernel is based on Gaussian function which can be written as,

$$K(x_i, x_j) = exp(-\frac{||x_i - x_j||^2}{2\sigma^2}) \tag{3.21}$$

where, x_i is the support vector and x_j is the current data value.

In representation of RBF kernel, $\frac{1}{2\sigma^2}$ is replaced with γ as shown in equation below.

$$K(x_i, x_j) = exp(-\gamma ||x_i - x_j||^2)\ for\ \gamma > 0 \tag{3.22}$$

The primary objective of this kernel is to build the decision boundary that classifies the training set into two separate parts. The mathematical representation of this decision boundary function using the RBF kernel is as follows.

$$f(x) = w.k(x_i, x_j) + b,\ w \in x_i, b \in \Re^m \tag{3.23}$$

where, b is the regularization parameter in m-dimensional feature space (represented by $\Re^m$).

The robustness of $f(x)$ can be ensured by minimizing the value of norm $||w||$ and by adding slack variables to counter the occurrence of inequalities in the classification problem [126]. The above decision function can now be represented as an optimization problem which is given by,

$$min \left(\underbrace{\frac{1}{2}||w||^2}_{Max.\ margin} - \underbrace{C\sum_{i=1}^{n}\xi_i}_{Min.\ error} \right) \tag{3.24}$$

The above equation can be solved by using Lagrange multipliers (α_i,$\beta_i \geq 0$) and the Eq. (3.23) can be represented as below.

$$f(x) = \sum_{i=1}^{n}(\alpha_i - \beta_i)K(x_i, x_j) + b,\ \alpha_i, \beta_i \in [0, C] \tag{3.25}$$

Using Lagrange multipliers in the above equation, the final objective function is represented as,

$$max \left(\sum_{i=1}^{n}\alpha_i - \frac{1}{2}\alpha_i\alpha_j u_i y_j K(x_i, x_j) \right) \tag{3.26}$$

$$subject\ to$$

$$\sum_{i=1}^{n}\alpha_i y_i = 0, \alpha_i \in [0, C] \tag{3.27}$$

3. Other kernels: There are many other kernels that can be used for classification of non-linear data such as-polynomial and chi-square kernels. Polynomial kernel uses a polynomial function as a kernel function which is shown in the below mentioned equation.

$$K(x_i, x_j) = (x_i^T x_j + c)^d \tag{3.28}$$

where c is the constant and d is the degree of polynomial function.

Chi-square kernels are based on the Chi-square distribution and are mathematically represented as shown below.

$$K(x_i, x_j) = 1 - \sum_{i=1}^{n} \frac{(x_i - x_j)^2}{\frac{1}{2}(x_i + x_j)} \tag{3.29}$$

Other kernels that can be used in variety of machine learning applications for classification of data can be found in [127]. However, considering the suitability of the problem, the Gaussian RBF kernel function is used in the proposed scheme.

3.4.4 Choose modeling parameters

For RBF kernel, two modeling parameters namely C and γ needs to be chosen before using RBF as a kernel function. These two parameters have a direct affect on the accuracy of the SVM classifier, thus selecting the right values for these parameters is an important task. In the proposed scheme, LIBSVM is used for optimizing these parameters on the basis of cross-validation and grid search [125].

In cross-validation, the training set is divided into various folds (say k), out of which, randomly chosen $k-1$ subsets are used for training the classifier and the remaining one is used for testing. Then, grid-search method is used for finding the values of C and γ [128]. Firstly, loose grid search with cross-validation is performed with respect to C and γ for values $(2^{-5}, 2^{-3}, ..., 2^{15})$ and $(2^{-15}, 2^{-13}, ..., 2^{3})$ respectively. Then, a finer grid search in the neighborhood of the (C, γ) pair (which gives the best cross-validation accuracy during the loose search) is performed to get the best values of C and γ. The classifier is again trained on the basis of the pair which gives the best cross-validation accuracy. The detailed description of the working of grid-search and cross-validation is described in [128].

3.4.5 Train and test the classifier

After the selection of kernel and modeling parameters, the SVM classifier is trained on the training set. Once the classifier is trained, then it becomes capable of classifying the new test values based on the trained classification model.

3.5 Performance Evaluation for Theft Detection

This section presents the simulation and results of the proposed scheme. In this scheme, two classifiers namely–SVM and DT are integrated together for the purpose of theft detection. Along with this, varied features have been chosen in order to relate to real-time scenarios. These features are namely–number of persons, number of appliances (with energy rating $> \phi$), temperature, time of day, season, actual electricity consumption and PE. In order to validate the working of the proposed scheme, energy consumption dataset for various homes in USA has been taken [129]. This dataset contained aggregated electric consumption data of these homes with respect to date and time. Using this dataset, other features considered in this work have been extracted from various sources and are linked together. Details about the features such as-temperature and season with respect to date and time have been obtained from historical meteorological data [130]. As the considered dataset did not contain any details about the number of persons in a particular house, therefore, average number of persons have been considered in each house. This is in accordance with the data published by USA government [131] and the average electricity consumption of the respective house. Following case study illustrates the working of the proposed scheme in detail.

3.5.1 Case study

For the sake of simplicity, this case study presents the scale-down version of the proposed system. In this case study, the collected dataset is assumed to represent the electricity consumption profiles of various homes scattered across a particular city. Apart from this, the considered data set is accommodated with 20% malicious data entries to represent electric theft across the city.

For the purpose of simulating this case study, different set of assumptions have been considered. These assumptions are mentioned as follows:

- A distribution station with the capacity of 2MW has been considered as power generation (PG) unit. This is in accordance with aggregated electricity consumption across the considered city and T&D losses.

- A typical electric distribution network of the considered city is illustrated in Fig. 3.7. As evident from the figure, the city comprises of a PG unit which is connected to two transmission lines (TL) and each TL further to two distribution stations (DS). Each of the DS is further connected to two transformers (T), which provide electricity to different localities. Each locality is assumed to be comprising of 140 houses.

- PG unit of 2MW capacity supplies 1MW power to each of the connected TLs. These TLs in return redistribute their power amongst the DSs. Hence, each DS receives 500kW power, which is further redistributed amongst the Ts (250kW) without considering the losses.

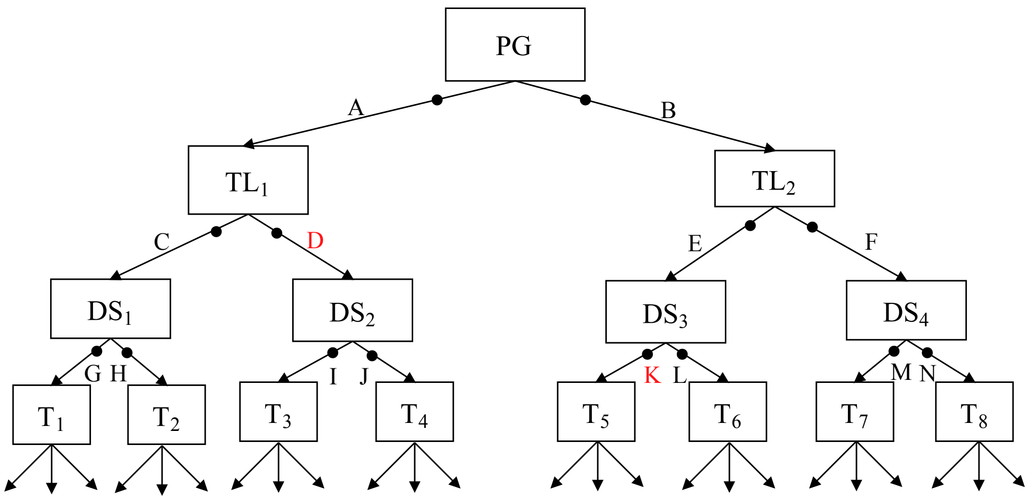


Figure 3.7: Scenario considered for case study.

- The total T&D losses (α) in the considered case study are assumed to be 5%. These losses are segregated across the system as shown in Fig. 3.7 and are in accordance with Eq. (3.1). The T&D losses from PG to TLs, TLs to DSs and from DSs to Ts are assumed to be 0.5%, 0.5% and 0.25% respectively. The losses from Ts to end users are considered to be negligible.

Electrical theft can take place at any level as discussed above. However, electrical theft at consumer (or end user) level is more profound in power systems. These thefts are typically concentrated across electricity transmission from Ts to the end users' premises. The electrical theft across all levels is detected using the proposed top-down approach and the related results at different levels, i.e, (i) prior to consumer level and (ii) at consumer level are summarized in Sections 3.5.2 and 3.5.3 respectively.

3.5.2 Theft detection prior to consumer level

Let us assume, all the electrical lines $\{A, B, ..., N\}$ as depicted in Fig. 3.7 are equipped with senors (S_i) where i is the i^{th} electric line. These sensors are placed at the beginning of their respective lines and are depicted by dot ($\bullet$) in the figure. Say, power tapping is being observed on electric lines D and K. In order to identify these thefts, sensors deployed across the system play a vital role, as they send the actual data related to electricity transmission to the utility server. The data (in kW) received by the utility server from various entities and sensors (shown in Fig. 3.7) is tabulated in Table 3.2.

As evident from Table 3.2, sensors S_A and S_B across the electric lines A and B report 1MW power through their respective lines. Since, the T&D losses at transmission level are assumed to be 0.5%, therefore, TL_1 and TL_2 should ideally receive 995kW of power. The data acquired by the utility server from TL_1 and TL_2 is also 995kW. Thus, it can be concluded that power transmissions across TL_1 and TL_2 are theft free. The same approach is propagated across all the levels one

Table 3.2: Data received by utility server from various entities.

Entity	PG	S_A	S_B	TL_1	TL_2	S_C
Value	2000	1000	1000	995	995	497.5
Entity	S_D	S_E	S_F	DS_1	DS_2	DS_3
Value	497.5	497.5	497.5	472.62	465	472.62
Entity	DS_4	S_G	S_H	S_I	S_J	S_K
Value	472.62	236.31	236.31	232.5	232.5	236.31
Entity	S_L	S_M	S_N	T_1	T_2	T_3
Value	236.31	236.31	236.31	230.4	230.4	226.28
Entity	T_4	T_5	T_6	T_7	T_8	
Value	226.28	222	230.4	230.4	230.4	

by one. For example, power transmission across electric line D is found to be theft prone, since the power received by DS_2 does not comply with the power transmitted by TL_1 after considering T&D losses. Similarly, electric line K is also found to be susceptible to theft, where power received by T_5 is 222kW against 230.4kW which should be ideally received by T_5.

3.5.3 Theft detection at consumer level

The proposed scheme exploits the advantages of DT-based approach combined with SVM classifier to identify electrical theft at the consumer level. Initially, PE is calculated using DT-based approach based on the features of consumers. For constructing the DT, E_x and IG are calculated using eqs. (3.2)-(3.4) for all features. The feature having maximum IG is selected as the parent node and the process continues until all the features are classified as child nodes. After this, priorities are assigned according to algorithm 2 based on the occurrence of features in the newly constructed DT. Depending on their respective priorities, individual weights are assigned to all the features. According to these assigned weights, PE for a consumer is calculated using Eq. (3.8). Likewise, PEs are computed for all the consumers and their respective PE values along with actual consumptions are illustrated in Fig. 3.8. As evident from this figure, there are several consumers whose actual electricity

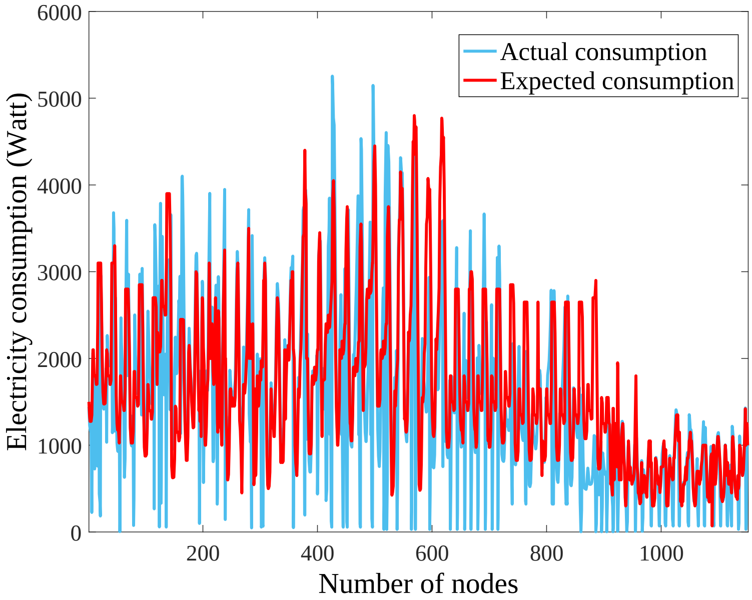


Figure 3.8: Expected and actual consumption of all consumers.

consumptions are significantly less than their respective PE values. This is because of the inclusion of both normal and malicious nodes in the analysis. To provide more clarity, two different graphs are presented for both types of consumers in Figs. 3.9 and 3.10. It can be inferred from these figures that PE of normal consumer fall within 10% range of actual consumption whereas, significant difference between PE and actual consumption is observed for the malicious ones. From the above analysis, it can concluded that PE plays a significant role in differentiating the malicious consumers. Hence, it acts as a good parameter for the proposed SVM classifier.

Once PE is computed, then SVM is used to classify the consumers either as malicious or normal. For this purpose, SVM accepts PE as one of the features along with other features, i.e., number of persons, number of appliances, temperature, time of day, season and actual consumption. The SVM classification is done in various steps as illustrated in Fig. 3.6. First step refers to pre-processing in which the data is converted into SVM compatible format. In the considered dataset, a feature named 'season' is in categorical form (having values such as-winter, spring, etc.) which is converted into numerical format.

The next step is to perform data normalization. For this task, values of each

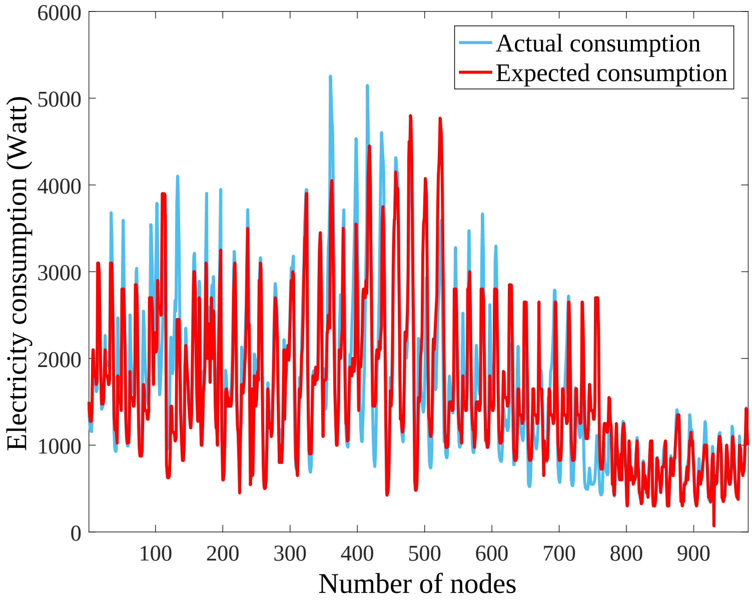


Figure 3.9: Expected and actual consumption of normal consumers.

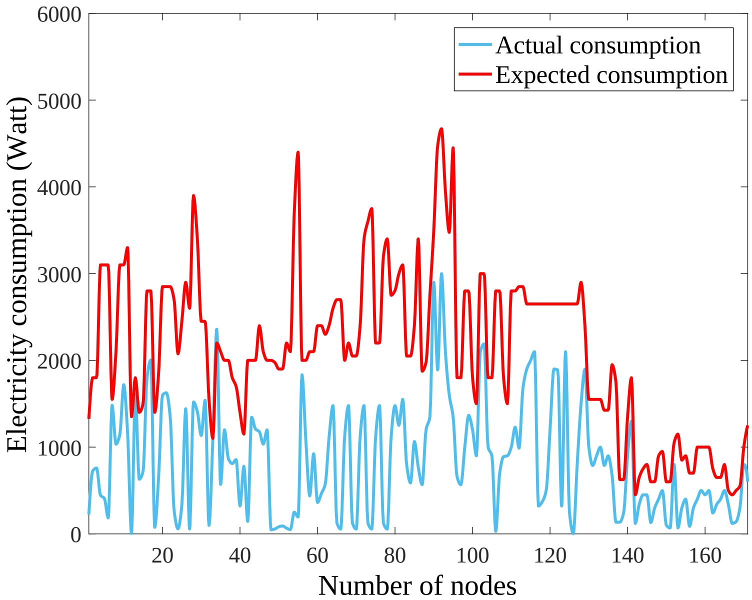


Figure 3.10: Expected and actual consumption of malicious consumers.

and every feature are computed using Eq. (3.9). For example, the number of persons and temperature in the collected dataset range from 2 to 6 and -33 to 33 respectively. These values are converted into [-1,1] range. The next step is to choose an appropriate kernel along with the modeling parameters. As already discussed in Section 3.4, the RBF kernel with modeling parameters C and γ is chosen in the proposed scheme. To compute values of C and γ, a loose grid search with 10-fold cross-validation is performed and the respective values are found to be 2 and 1. The fine grained grid search is then performed in the proximity of (2,1) for computing optimal values of C and γ. The associated results are shown in Fig. 3.11. The values of (C,γ) which exhibits maximum 10-fold cross-validation accuracy are found to be (1.96,0.98). The next step is to train the classifier using these values.

Initially, the SVM is trained on the basis of five parameters mentioned earlier. In order to test the accuracy of the proposed SVM classifier, the ratio of training and testing data is taken as 70-30. Once the SVM classifier is trained, it is evaluated on the test data to calculate the detection rate. This classifier gives the accuracy of 95.5% on the training data and detection rate of 87.5% for the test data. It is to be noted that different results would be obtained if the ratio of training and test dataset varies. The results on different percentages of training set and the associated test accuracies are shown in Fig. 3.12 along with another case which is discussed below.

Another case is also considered in this work, where the SVM classifier is trained along with DT's output. In this case, the training and testing accuracies of SVM classifier increases upto 97.5% and 92.5% respectively. The training accuracies are

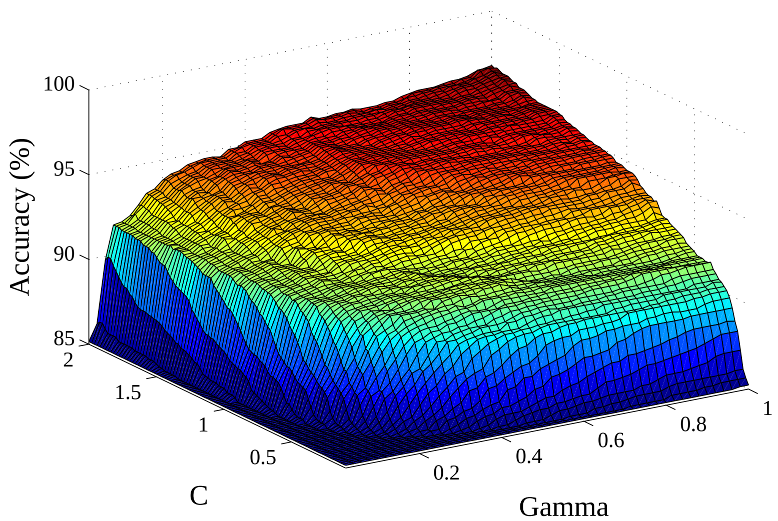


Figure 3.11: Grid search cross-validation accuracy variations.

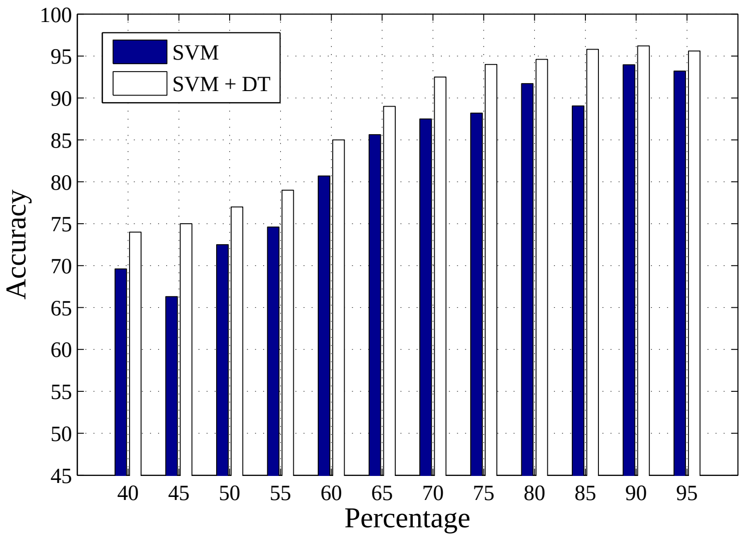


Figure 3.12: Accuracy with varying percentage of training set.

taken after training the SVM classifier by considering 10-fold cross-validation. The results for other cross-validation accuracies for both the cases are given in Fig. 3.13.

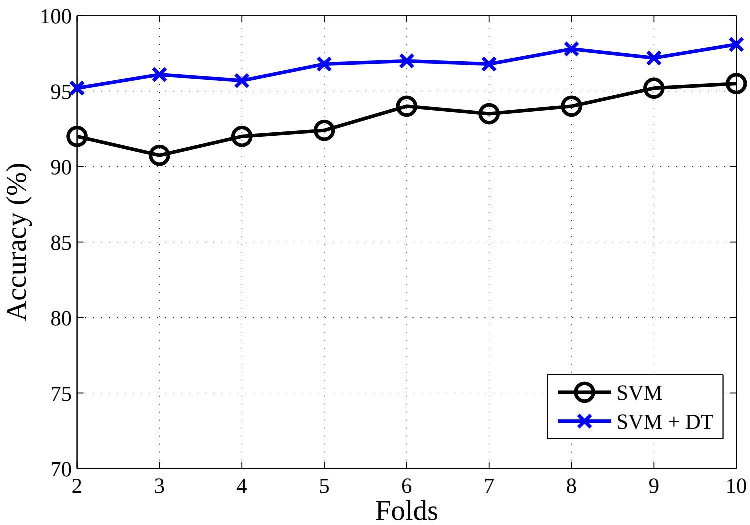


Figure 3.13: Cross-validation accuracy.

This proves that the obtained results are better if DT is used in conjunction with SVM classifier. Furthermore, the false positive rate in this case is also very less, i.e., 5.12% which is acceptable in practical scenarios. Thus, it can be inferred that the proposed scheme is reliable enough to be implemented in the realistic environment for the purpose of detecting thefts in the electrical networks.

Apart from the above analysis, it is quite essential to compute the response time of the proposed scheme in order to process the data of the homes for theft detection. The utility server processes simultaneous requests (i.e., incoming data of homes) and the related response times have been depicted in Fig. 3.14. However, the communication delay involved in relaying the consumption data from smart meters to utility has not been considered. This analysis has been carried out using Intel Core i7-2640M processor operating at 2.8GHz with 4GB of primary memory. The results clearly indicate that the proposed scheme has low response time and is capable to address multiple requests at ease.

3.5.4 Implementation in a practical scenario

In this segment, a practical scenario has been considered to detect the electricity theft at consumer level using partial deployment of smart meters. As shown in Fig. 3.15, the considered setup consists of 5 blocks comprising of 24 homes each, where h_{ij} represents the j^{th} home of the i^{th} block. Out of these homes, only 20% are facilitated with smart meters while others have conventional digital meters. Thus,

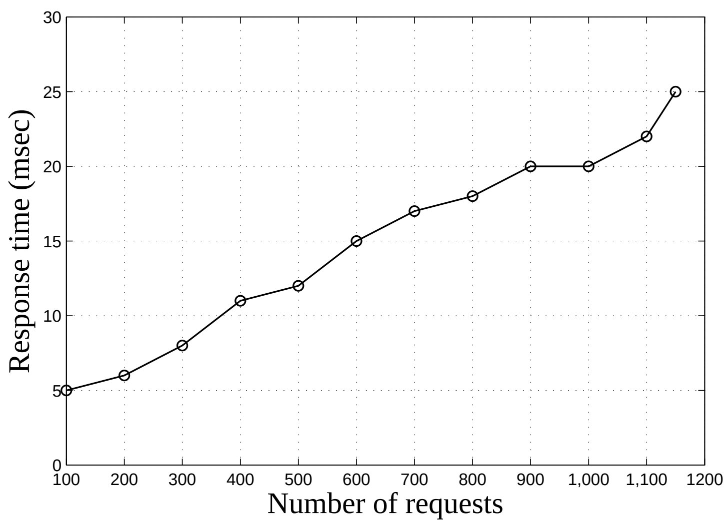


Figure 3.14: Response time of the proposed scheme.

only 5 homes in each block are equipped with smart meters. As depicted in the figure, the homes in all the blocks are powered through their respective electric poles P_A, P_B, P_C, P_D, and P_E. Apart from this, it is considered that a total of 3 homes (2 from block A and 1 from block C) are involved in drawing illegal and extraneous power.

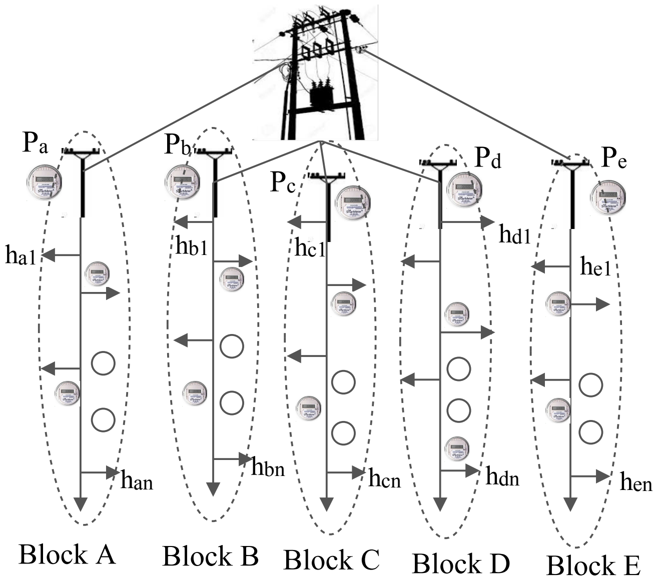


Figure 3.15: Practical scenario with limited smart meters.

The electricity theft in homes equipped with smart meters are detected using the proposed scheme as already discussed in Section 3.5.3. For rest of the homes, the proposed SVM based approach cannot be used as it requires actual electricity consumption data from smart meters for theft analysis. In such scenarios, DT is used for theft detection and it works in the following manner. Initially, the PE consumption values of all the homes are estimated using Eq. (3.8) as already discussed in Section 3.3. For this purpose, DT utilizes the input parameters already available at the utility's end as shown in Fig. 3.3. The estimated PE values for the considered scenario are illustrated using Fig. 3.16.

After this step, PE consumption (PE_{ij}) of all homes without smart meters is computed using DT. On the other hand, the actual consumption (A_{ij}) data of homes equipped with smart meters is acquired. Once this data is gathered, then aggregated

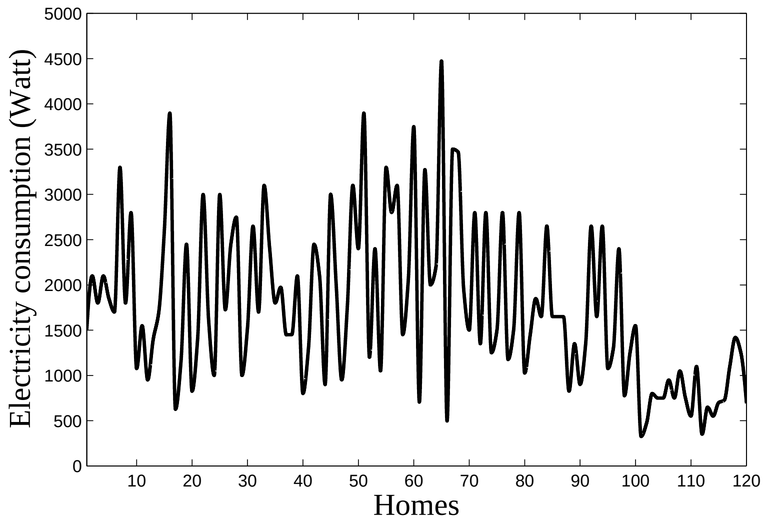


Figure 3.16: Expected electric consumption of the individual homes.

consumption (Agg_i) of each block is estimated using the below mentioned equation.

$$Agg_i = PE_{ij} + A_{ij} \tag{3.30}$$

The value of Agg_i for each block is then compared with the electric power transmitted from their respective pole P_i. In case, Agg_i turns to be less than value recorded by pole P_i, then the considered i^{th} block is assumed to be under potential electric theft. Otherwise, it can be inferred that there is no theft in the block.

Let us consider the practical setup as shown in Fig. 3.15 and extend the above mentioned approach to detect the fraudulent customers. In block A, the power at pole P_a at an instant is 98 kW whereas, Agg_a of all the homes is merely 87 kW. This signifies that block A is under electric theft. Similarly, block C is also affected by theft where the value at pole P_c and Agg_c are 104 and 95 kW respectively. In rest of the blocks, little variations are observed between values at poles P_i and estimated Agg_i values as depicted in Fig. 3.17. Thus, it can be concluded that blocks B, D, and E are free from electric thefts and the adopted DT based approach is in-line with the proposed scheme.

These results clearly indicates that this scheme is capable enough to identify

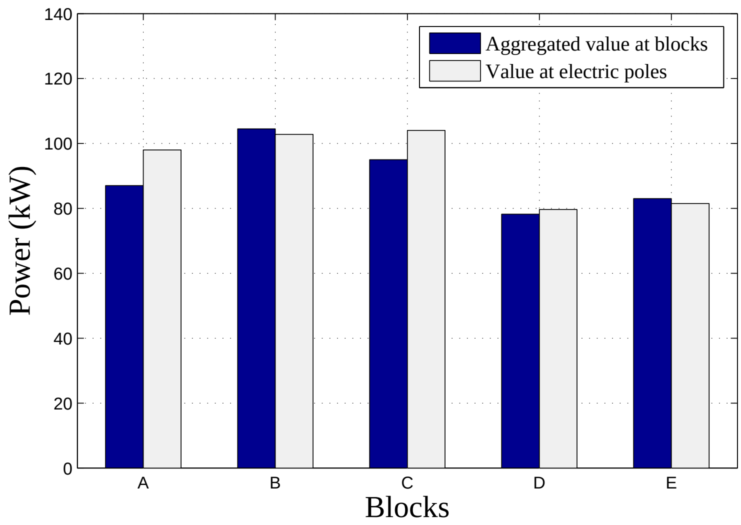


Figure 3.17: Aggregated consumption values of blocks and metered values at poles.

electricity theft even under partial deployment of smart meters. Thus, the proposed scheme can work well in the real world scenario.

3.5.5 Comparison with existing schemes

This segment elaborates the comparison of the proposed scheme with the existing schemes on the basis of three evaluation parameters namely-accuracy, false positives and approach used. The comparison results summarized in Table 3.3 reveal that the proposed scheme is the most effective scheme relative to others [58–61, 64]. For instance, the accuracy of the scheme is as high as 92.50% while the accuracy of the other schemes merely lie between 20-75%. Furthermore, the proposed scheme offers very limited false positives due to inclusion of both DT and SVM based approaches. On the other hand, the schemes presented in [58, 60, 61] lead to high false positive rates. In addition to this, the proposed scheme helps to identify thefts at both transmission and consumer levels. All the other schemes primarily concentrate on detecting the thefts at consumer level only. Thus, it can be concluded from the detailed comparisons that the proposed scheme is better than others and provides higher accuracy for detecting the electric thefts.

Table 3.3: Comparative analysis of the proposed theft detection scheme.

Scheme	**Technique adopted**	**Accuracy**	**False positive**	**Approach used**
Angelos *et al.* [59]	Fuzzy classification	74.50%	Medium	Consumer level
Cabral *et al.* [60]	Rough sets	20%	Very High	Consumer level
Costa *et al.* [58]	ANN	65.03%	High	Consumer level
Nagi *et al.* [61]	SVM	60%	High	Consumer level
Nagi *et al.* [64]	Fuzzy logic coupled with SVM	72%	Medium	Consumer level
Proposed scheme	DT coupled SVM	92.50%	Very Low (5.12%)	Transmission and consumer level

3.6 Summary

Electrical theft leads to enormous losses to the utilities in the power sector. The major cause of these thefts is the illegal use of electricity by the consumers through tapping. To detect the malicious consumers that intentionally steal the electricity, a novel top-down scheme has been presented in this chapter using DT and SVM classifiers. The proposed scheme is capable enough to detect the thefts happening anywhere in the power network. To detect the malicious consumers, various features namely number of heavy appliances, number of persons, season, time slot and temperature are given as input to the DT. The expected electricity consumption for the consumers during a particular time is calculated using DT. This consumption along with other features is provided as input to the SVM classifier which is trained on the collected dataset. This classifier is then used to classify the consumers as normal or fraud based upon their features. Results prove that the proposed scheme identifies fraudulent consumers with an accuracy of 92.5% and a false positive rate as low as 5.12%. This scheme also works well in a scenario where a very less number of smart meters are deployed. Results also prove that the accuracy of SVM classifier is improved when it works in conjunction with DT.

Chapter 4

Demand Response Management

There are various issues such as–increasing demand-supply gap, grid instability and deteriorating quality of service that persist in the energy network which degrade its performance. These issues can be handled in an efficient way by managing the demand response of different types of loads such as–smart homes and EVs/PHEVs. For this purpose, a novel scheme to handle the demand response of smart homes and PHEVs is presented in this chapter[1]. This scheme is based on analyzing the demand requirements of these users and is divided into two hierarchical stages which work as follows. In the first stage, the residential and PHEV users are identified whose demands can be regulated. This task is achieved with the help of a binary-class SVM which uses Gaussian kernel function to classify these users. In the next stage, load in smart homes is curtailed on the basis of a pre-defined rule-base after analyzing the consumption data of various devices; whereas PHEVs are managed by controlling their charging rates. Results prove that the proposed scheme is able to flatten the load profile of smart grid to a great extent by managing the demand

[1]The contents of this chapter are partly published in:

- A. Jindal, N. Kumar, and M. Singh, "Internet of Energy-based Demand Response Management Scheme for Smart Homes and PHEVs Using SVM," *Future Generation Computer Systems*, 2018, DOI: 10.1016/j.future.2018.04.003.
- A. Jindal, N. Kumar, and M. Singh, "A Data Analytical Approach Using Support Vector Machine for Demand Response Management in Smart Grid," in *IEEE PES General Meeting*, Boston, MA, Jul. 2016, pp. 1-5.

response of residential sector and PHEVs.

4.1 Contributions

The major contributions of this chapter are summarized as follows.

1. It provides a unified solution to manage the demand response of residential loads and PHEVs in an energy network by analyzing their demand requirements.

2. A load balancing algorithm has been designed to curtail the load of residential users (identified using SVM classifier) on the basis of a pre-defined rule-base by analyzing their consumption data.

3. A binary-class SVM classifier has been used to identify the PHEVs whose charging rates can be altered in order to flatten the load on smart grid.

4.2 Proposed Scheme

The proposed scheme deals with managing different types of loads in an energy network on-the-fly for each time slot. These loads belong to various categories viz. industrial & commercial loads, residential loads and load due of PHEVs as shown in Fig. 4.1. In the proposed scheme, industrial and commercial load demands are not altered as any modification made in their demand can severely affect their business and can cause huge economic losses. As far as the other load categories are concerned, i.e., residential load (comprising of smart homes) and PHEVs, there demand response is managed in the proposed scheme to flatten the load profile of the grid. The consumption data related to these load categories is gathered at the utility server as shown in Fig. 4.2. The PHEVs can be charged in the smart homes or at charging stations (CSs). The proposed scheme follows different demand response management strategy to handle the charging requirements of PHEVs at these places.

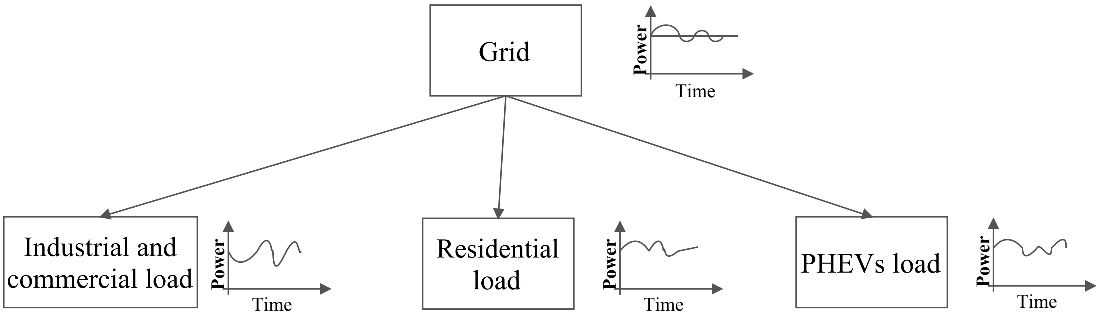


Figure 4.1: Various types of load categories.

In the smart homes, the charging of PHEVs is re-scheduled; while at the CSs, the charging rates of PHEVs are altered. In order for the proposed scheme to work properly, the existence of two conditions is essential. Firstly, the users are willing to participate in the proposed demand response management program. Secondly, the PHEVs present at CSs and devices in smart homes are able to communicate their usage data to a centralized control entity. For the first condition, additional benefits such as–reduced electricity tariff, cash incentives, etc. can be provided to the users; while the second condition can be satisfied with advancements in ICT. Thus, it is assumed that these two conditions remain persistent throughout the proposed scheme.

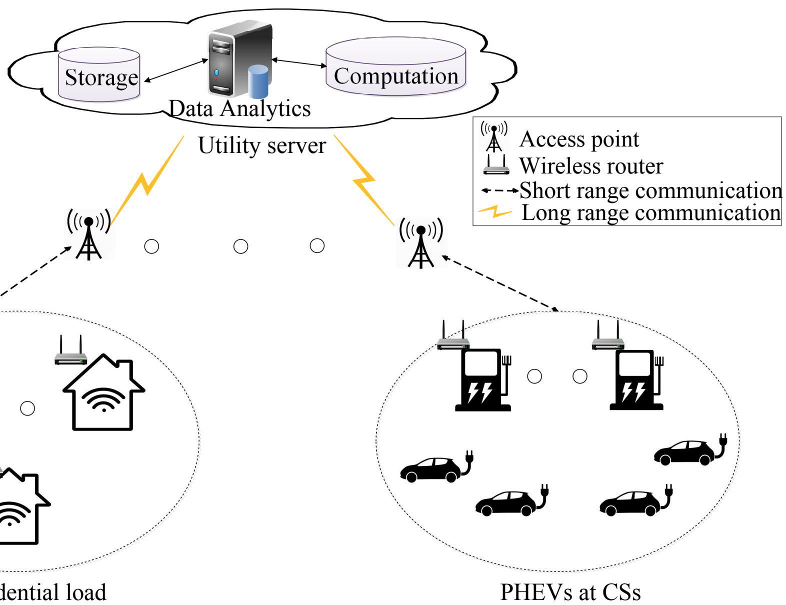


Figure 4.2: Communication layout between users and utility server.

The complete load management approach is dependent on analyzing the usage data gathered from various devices in the large energy network. The usage data of smart devices present in smart homes is gathered at the local controller using ZigBee, Wi-Fi, IEEE 802.15.4a/c/f or other related technologies which is then sent to the utility server where the overall demand profile of a smart home is generated. The PHEVs' demand data is gathered when they plug-in at the CSs [132]. This data is then relayed to the server with the help of access points (APs) which make use of various technologies such as–IEEE 802.11p, Wi-Fi and WiMAX for bi-directional communication as shown in Fig. 4.2. Once the data is gathered at the utility server, it is analyzed as explained below.

Initially, the total load at smart grid is computed from the energy consumption data of PHEVs and devices present in smart homes. This load is then compared with generation capacity of the grid and if there is a peak in demand, the loads are managed using data analytics as follows. Initially, the SVM classifier is modeled to identify the residential users with excess load consumption; the load of such users is then curtailed using a load balancing algorithm. If the load profile of grid is still not flattened, then another SVM classifier discovers the suitable PHEVs placed at CSs whose charging rates can be regulated. The charging rates of these PHEVs are then altered to manage the demand at smart grid.

In case of valley filling, the loads are managed in the following way. Initially, the devices in smart homes (that were switched OFF during peak shaving) are switched ON. If the overall demand is still less than the power generation, then the charging rates of PHEVs (identified by the SVM classifier) are increased to fill the valley. In this way, the demand response of smart homes and PHEVs is managed by the proposed scheme. The detailed working of this scheme for each type of load is explained in the subsequent sections.

4.3 Residential Load Management

The problem of managing the demand response of residential load has been divided into two phases. In the first phase, SVM classifier bifurcates the smart homes into two classes, i.e., smart homes with normal load consumption and excess load consumption. In the next phase, the demand response of smart homes (labeled with excess load consumption) is handled using a novel load balancing algorithm on the basis of their consumption data and rule-base. The working of these phases is shown in Fig. 4.3 and elaborated as follows.

4.3.1 Identifying smart homes with excess load consumption

A binary-class SVM classification is being used to label the smart homes which exceed the normal load consumption. SVM has been extensively used in many related applications because it is computationally efficient and has high classification accuracy [133,134]. The importance of using SVM in the proposed demand response management process is that it compares the energy consumption pattern of all the smart homes to assign labels. The comparison for energy consumption of different smart homes using SVM is made on the basis of various parameters and is relative in nature. For example, if one smart home is consuming more energy than other smart homes (based on classification parameters), then it is labeled as smart home with excess load consumption. For this purpose, the usage data of smart homes is sent to the utility server as shown in Fig. 4.2. The parameters used in this classification are season, temperature, time slot, present load consumption and number of members in a household. The steps used in the SVM classification are shown in Fig. 4.3 and are explained below.

1. **Data pre-processing:** In this step, the categorical data gathered from smart homes is transformed into numeric format as required in the SVM classification. A dataset is then prepared which stores the data related to classification parameters

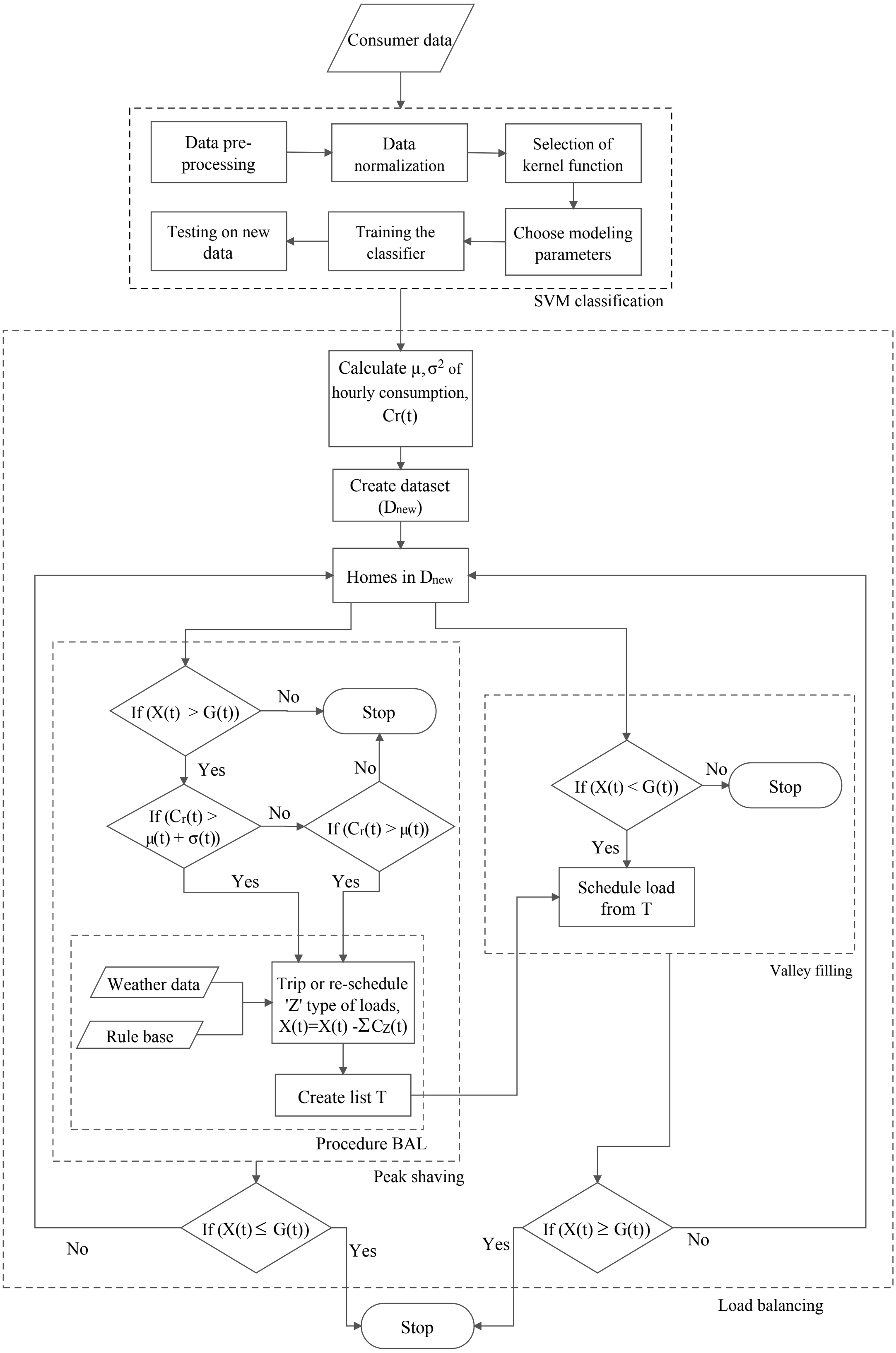


Figure 4.3: Phases in residential load management.

of all the smart homes. A unique identity number is associated with every device of a smart home to ensure the correctness of data and to remove any uncertainties in the system while preparing the dataset.

2. Data normalization: Before SVM classification is carried out, the data needs to be normalized in order to give equal weightage to all the considered features. The normalization is performed using the equation given below.

$$V' = \frac{V - V_{min}}{V_{max} - V_{min}} \tag{4.1}$$

where, V' is the normalized value of the feature that lies between [0,1] and V is the feature value that is to be normalized. V_{min} and V_{max} are the minimum and maximum values of variable V in the given dataset respectively.

3. Selection of kernel function:

Another important factor associated with the SVM classification is the selection of kernel function. Different kernel functions can be applied on the data such as–linear and non-linear, depending on the type of data. When the solution set is linearly separable or when number of features are large and dataset is small, then linear kernel is used. For all the other cases, non-linear kernels are used and most commonly used non-linear kernel is the Gaussian kernel. As the dataset used in the proposed scheme is non-linearly distributed over a small feature set, thus, the Gaussian kernel is used for classification purpose which is mathematically represented as,

$$\ker(x_i, y_i) = exp(-\gamma(x_i - y_i)^2) \tag{4.2}$$

where, $\ker(x_i, y_i)$ is the kernel function, x_i is the support vector, y_i is the considered data value and γ is the free parameter that decides the spread of kernel function. The above function forms a bell-shaped curve and the width of bell is dependent on the value of γ; larger values of γ yield narrower curves and vice-versa.

The function of this kernel is to create a decision boundary (or hyperplane) which can classify the training set into two separate classes. Using this kernel function, the decision boundary equation is given by,

$$f(x_i) = w.\ker(x_i, y_i) + b \tag{4.3}$$

where, w and b represents the normal vector to the plane and regularization parameter, respectively.

To counter the effect of inequalities in the above function, slack variables (ζ) are included in the classification problem [134]. This problem is now represented as an optimization problem which is given below.

$$min \left(\underbrace{\frac{1}{2}||w||^2}_{Max.\ margin} + \underbrace{C\sum_{i=1}^{n}\zeta_i}_{Min.\ error} \right) \tag{4.4}$$

$$subject\ to$$

$$y_i(w.ker(x_i, y_i) + b) \geq 1 - \zeta_i,\ \forall\ i \tag{4.5}$$

$$\zeta_i \geq 0,\ \forall\ i \tag{4.6}$$

Equation (4.4) is a convex optimization problem where C is the control parameter used for trading-off between maximum margin and minimum error. This equation helps to solve the under and over-fitting problem of training data by minimizing the values of $\frac{1}{2}||w||^2$ and the training error ($C\sum_{i=1}^{n}\zeta_i$) [134]. Equation (4.4) is solved with the help of Lagrange multipliers and the final objective function is changed to,

$$f(x) = \sum_{i=1}^{n}(\alpha - \beta)\ker(x_i, y_i) + b;\ \alpha, \beta \in [0, C] \tag{4.7}$$

where, α and β are Lagrange multipliers.

4. Optimizing modeling parameters:

The modeling parameters C and γ are also required in the Gaussian kernel function. The value of these parameters needs to be optimized for greater accuracy. To compute the optimized values of C and γ, LIBSVM [125] software package is used by implementing grid search and cross validation methods [128].

5. Training and testing of classifier:

The final classifier is trained on the input dataset using Gaussian kernel function on the optimized values of C and γ. The trained classifier is now capable of classifying any new smart home data gathered in smart grid. The data of homes that are labeled as excess load consumption by this classifier is stored in dataset D_{new}. This dataset is then fed to a load balancing algorithm for curtailment of load in smart homes as discussed in the following segment. To adapt the SVM classifier to the dynamically changing conditions in smart homes, it is re-trained at regular intervals by adding new data values collected from smart homes.

4.3.2 Balancing the excess load consumption in smart homes

Once the smart homes with excess load consumption are identified using SVM classifier, algorithm 3 is executed at the utility server for each time slot. In this algorithm, the dataset D_{new}, total load at smart grid at t, $X(t)$, and its generation capacity, $G(t)$, are provided as input. The working of this algorithm is depicted in Fig. 4.3 and is explained below.

The hourly mean, $\mu(t)$, and standard deviation, $\sigma(t)$, in electricity consumption of smart homes are calculated from past data (considered for the previous week) and are stored in D_{new}. The value of $C_r(t)$, i.e., total electricity usage in a home at t^{th} time slot, is calculated from the device data and stored in D_{new} (lines 1-4). These homes are then sorted according to their decreasing values of $\sigma(t)^2$ (line 5). When the grid faces peak in demand (i.e., $X(t) > G(t)$), a function BAL (explained below)

Algorithm 3 Balancing residential load demand

Input: D_{new}, $X(t)$, $G(t)$

Output: Balanced load demand, T

```
for (Each home in D_new) do
    Calculate hourly μ(t), σ(t) from historical data
    Calculate total electricity usage (C_r(t)) and store in D_new
end for
Sort D_new according to decreasing value of σ(t)^2
if (X(t) > G(t)) then                                          ▷ Peak shaving
    while (X(t) ≤ G(t)) do
        For homes in D_new
        if (C_r(t) > μ(t) + σ(t)) then
            BAL(C_r(t), μ(t), σ(t))
        end if
        if (D_new == empty where C_r(t) > μ(t) + σ(t)) then
            for (i = 1; i ≤ n, i++) do                         ▷ n ← Total smart homes
                Find smart homes where C_r(t) > μ(t) & store their data in D_ex
            end for
        end if
        for (Homes ∈ D_ex) do
            BAL(C_r(t), μ(t), 0)
        end for
    end while
end if
if (X(t) < G(t)) then                                          ▷ Valley filling
    while (X(t) ≤ G(t)) do
        Pick appliances from list T and switch them ON
    end while
end if
                                                               ▷ Instantaneous load change
When any new device (with load L_new(t)) is switched ON
if (X(t) + L_new(t) < G(t)) then
    Switch ON the device
else
    Consider that device for next time slot
end if
When any existing device (with load L_old(t)) is switched OFF
if (X(t) − L_old(t) < G(t)) then
    Switch ON load from T until X(t) ≤ G(t)
else
    do nothing;
end if
 Procedure BAL(C_r(t), μ(t), σ(t))
while (C_r(t) ≤ μ(t) + σ(t)) do
    Trip loads according to rule-base and save these in list T
    X(t) = X(t) − ∑ C_z(t)                                     ▷ C_z ← Load of tripped device z
end while
return X
end procedure
```

is used to reduce the load of homes having $C_r(t) > (\mu(t) + \sigma(t))$ (lines 6-11). If no such smart home is present in D_{new}, then homes having $C_r(t) > \mu(t)$ are identified and their data is stored in new dataset, D_{ex} (lines 12-16). The function BAL is then applied on such homes to manage their load demand (lines 17-19). The above procedure is repeated until $X(t) \leq G(t)$ (line 7 to 20). On the other hand, during valley filling, the appliances from list T (created in BAL function) are switched ON until the demand and supply gap is minimized (lines 22-26). The algorithm is also capable of managing the instantaneous changes in residential load. Whenever a new device (with load L_{new}) is switched ON, the utility would accommodate that device, if $(X(t) + L_{new}(t)) < G(t)$ (lines 28-30). Otherwise, it would be considered for the next time slot (line 32). When an existing device (with load L_{old}) is switched OFF, the devices from list T are switched ON if the value of $(X(t) - L_{old}(t)) < G(t)$ (lines 34-39).

The function BAL plays a crucial role in peak shaving and valley filling. It re-schedules the residential load according to a pre-defined rule-base until $C_r(t) \leq (\mu(t) + \sigma(t))$. The cumulative value of these loads is then decreased from $X(t)$ and their information is stored in list T (lines 40-44). The rule-base used in function BAL decides the actions to be taken for different input conditions. These conditions depict the weather information and device status based on which various actions are specified. Few of such input conditions and their respective actions are given in Table 4.1. The comfort of the user is also taken into account while formulating the rules specified in rule-base. For example, if the input condition is <*Temperature is hot*>, then the action will be <*Switch OFF heater*>. In this case, switching OFF the heater would not reduce user's comfort level (as outside temperature is hot); however, the electricity consumed by the heater would be reduced from the overall load on the grid. In this way, data analytics is applied to trip or reschedule the load in a smart home without affecting the users' comfort.

It is important to mention here that the emergency devices such as fans, tube-lights, bulbs and devices related to health care, elderly care and infant care are kept

Table 4.1: An instance of the rule-base.

Condition	***Action***
Weather is cold	Switch OFF the AC
Temperature is hot	Switch OFF the heater
AC is ON for ‘t’ time slots	Switch OFF and schedule at ‘t+1’
Water heater is ON for ‘t’ time slots	Switch OFF and schedule at ‘t+1’
Washing machine is ON	Switch OFF and store in T
PHEV is charging	Switch OFF and store in T

out of residential demand response management process. It may be possible that after applying algorithm 3, the complete load profile of the grid is not flattened. This case arises when the input conditions of smart homes do not match with the conditions specified in the rule-base. In such scenarios, the load demand is handled using PHEVs as specified in upcoming section.

4.4 Managing PHEVs Load at CSs

The PHEVs placed at CSs are used to manage the load profile of smart grid when demand response management of smart homes is not sufficient to balance the gap between demand and supply. In this scenario, the PHEVs whose charging rates can be regulated (decreased while peak shaving and increased while valley filling) are identified with the help of SVM classifier. Many researchers have used SVM classification with respect to PHEVs for estimating their state of charge (SoC) [135] and state-of-health (SoH) [136] of battery. However, according to the best of our knowledge, no research work have focused on classification of PHEVs for determining their participation in the demand response management process.

For this purpose, two binary-class SVM classifiers are modeled for peak shaving and valley filling respectively. These classifiers assign the PHEVs into different classes on the basis of various input parameters as shown in Fig. 4.4. These classes are namely slow chargeable PHEVs (SCPs) & normal chargeable PHEVs (NCPs), in case of peak shaving and fast chargeable PHEVs (FCPs) & NCPs, in case of valley filling. The input parameters used for PHEVs classification are initial SoC

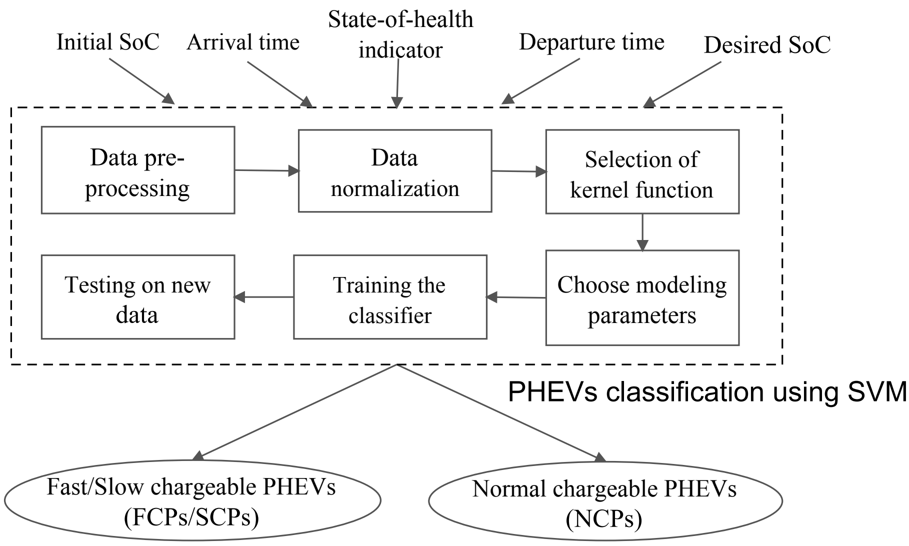


Figure 4.4: Classification of PHEVs using SVM.

level, desired SoC level, arrival time of PHEV, departure time of PHEV and SoH indicator of PHEV's battery. The significance of using SVM classification for PHEVs in the proposed scheme is that the charging rates of only those PHEVs which are labeled as FCPs or SCPs will be regulated. In this way, the PHEVs that may suffer battery degradation due to variation in charging rates would be kept out of demand response management process. One of the important parameters which is directly related to the battery degradation is 'SoH indicator'. It is defined as the measure of decay of battery due to capacity fading over the period of time. It is an important factor which needs to be considered while altering the charging rates of PHEV's battery (especially in case of fast charging). It is because the PHEVs with low value of SoH indicator would have higher charging losses and their charging rates should not be regulated. Moreover, altering the charging rates of such PHEVs may further deteriorate their battery capacity.

The SVM classifier models the relationship between the input factors to identify which PHEVs are best suited for regulating the charging rates. This relationship can be derived in terms of difference between departure time & arrival time, difference between desired SoC & initial SoC and SoH indicator of the battery. The data related to factors viz. initial SoC, arrival time, departure time and desired SoC is gathered when PHEVs' plug-in at CSs to charge their batteries. To calculate the value of SoH indicator of PHEV's battery, many techniques [136,137] and tests [138] are available in the literature. However, all these studies focus on laboratory testing rather than the real-time calculation in practical situations. Thus, for implementing the proposed scheme, a simple SoH indicator estimator has been used which is based on battery degradation as given in Eq. (4.8) [132].

$$B^i_{soh} = \frac{N^i_t - N^i_p}{N^i_t} \tag{4.8}$$

where, B^i_{soh}, N^i_p and N^i_t are the SoH indicator, present charging cycle and total number of charging cycles of the i^{th} PHEV's battery. The value of N^i_t is fixed and pre-defined by the manufacturer of the battery; whereas the value of N^i_p would be known to the user.

After gathering the data about these factors, SVM classifiers are trained in the same way as discussed in Section 3.4. The SVM training process in the case of PHEVs is given in algorithm 4. In this process, initially the gathered data is converted into numeric format as depicted in the data pre-processing step. Then, data normalization is applied using Eq. (4.1) to give equal importance to all the features. In the next steps, Gaussian kernel function is used to classify the data as given in Eq. (4.2) and values of modeling parameters C and γ are optimized using LIBSVM [125]. Finally, the SVM classifiers are trained on the basis of normalized input dataset.

Now, using the classifiers given by algorithm 4, the PHEVs are labeled as FCPs/SCPs or NCPs. These classifiers are trained such that the PHEVs which have good SoH indicator and require less SoC in more time are classified as FCPs during valley filling. Similarly, the PHEVs requiring more charge in less time with good value of SoH indicator are labeled as SCPs during peak shaving. The PHEVs having bad SoH indicator are labeled as NCPs to prevent them from further battery deterioration. For the PHEVs labeled as SCPs (or FCPs), the charging rate of their

Algorithm 4 Training SVM classifier for PHEV classification

Input: Input parameters of PHEVs
Output: SVM classification model

```
Convert feature data into numeric format                         ▷ Pre-processing
                                                                 ▷ Data normalization
for (i = 1; i ≤ n; i++) do                                       ▷ For every input parameter F_i
    Normalize all the data values of F_i in range of [0,1] using Eq. (4.1)
    Store the normalized values in a dataset D_n
end for
                                                                 ▷ Choosing the kernel function
Use Gaussian kernel as given in Eq. (4.2) as the kernel function
                                                                 ▷ Finding modeling parameters
Use grid search method on D_n to choose modeling parameters (C, γ)
Use cross-validation with grid search to find best C and γ
Use calculated C, γ to train the classifier on D_n
return SVMclassifier(PHEVs)
```

batteries during peak shaving (or valley filling) would be regulated as discussed in the following sections.

4.4.1 Peak shaving using PHEVs

For the PHEVs labeled as SCPs, the charging rates of such PHEVs' batteries are altered according to algorithm 5 until the load profile of smart grid is flattened.

Suppose that $P^{req}(t)$ is the power required by smart grid from CSs at t^{th} time slot during peak time. This power is drawn from all the CSs which is distributed amongst them according to equation given below.

$$P_{CS_i}^{draw}(t) = \frac{P_{CS}^{i}(t)}{P_{CS}(t)} P^{req}(t) \tag{4.9}$$

such that

$$\sum_{i=1}^{m} P_{CS_i}^{draw}(t) = P^{req}(t) \tag{4.10}$$

where, $P_{CS_i}^{draw}(t)$ is the power drawn from the i^{th} CS in the t^{th} time slot, $P_{CS}^{i}(t)$ is the total power requirement by i^{th} CS from smart grid and $P_{CS}(t)$ represents the total power requirements of all CSs. At a single CS, this required power is distributed proportionally amongst different SCPs at that CS according to the given equation.

$$P_{b_{ji}}^{draw}(t) = \frac{P_{b_j}^{i}(t)}{P_{CS}^{i}(t)} P_{CS_i}^{draw}(t) \tag{4.11}$$

where, $P_{b_j}^{i}(t)$ is the power of j^{th} SCP's battery at i^{th} CS and $P_{b_{ji}}^{draw}(t)$ is the power that is to be drawn from this PHEV. The charging rate of such PHEV batteries is managed according to the equation given below.

$$C_{ji}^{new} = C_{ji}\left(1 - \frac{P_{b_{ji}}^{draw}(t)}{P_{b_j}^{i}(t)}\right) \tag{4.12}$$

such that

$$\begin{cases} 0 < C_{ji}^{new} \leq 1; \\ SoC_{cur} \leq SoC_{des} \end{cases}$$

where, C_{ji}^{new} is the new charging rate of j^{th} SCP's battery, C_{ji} was its present charging rate, SoC_{cur} is the present SoC and SoC_{des} is the desired SoC level of PHEV's battery.

The instantaneous load change at the CS during peak shaving is handled as follows. When a new PHEV joins the i^{th} CS at any time (t^k), then the energy required by it is compensated locally by altering the charging rates of other SCPs' batteries present at this CS. For this purpose, the new value of $P_{CS}^{i}(t)$ is calculated at t^k and the value of $P_{b_{ji}}^{draw}(t)$ is updated for each SCP. The corresponding value of C_{ji}^{new} is then computed according to which the battery of each PHEV is charged. Similarly, when an existing PHEV leaves the CS at t^k, the charging rates of rest of SCPs are re-calculated according to Eq. (4.12).

Algorithm 5 Managing PHEVs load demand at CSs

Input: $P^{req}(t)$, $P^{ext}(t)$

Output: Balanced load demand

When smart grid needs to draw $P^{req}(t)$ from all the CSs ▷ **Peak shaving**
for $(i = 1; i \leq m; i++)$ **do** ▷ m←Total number of CSs
    Calculate power to be drawn from CS_i ($P^{draw}_{CS_i}(t)$) according to Eq. (4.9)
    for $(j = 1; j \leq n; j++)$ **do**
        Calculate power drawn from j^{th} SCP according to Eq. (4.11)
        Manage the charging rate of j^{th} PHEV according to Eq. (4.12)
    end for
end for
When smart grid needs to distribute $P^{exc}(t)$ to all the CSs ▷ **Valley filling**
for $(i = 1; i \leq m; i++)$ **do**
    Calculate excess power supplied to CS_i ($P^{excess}_{CS_i}(t)$) according to Eq. (4.13)
    for $(j = 1; j \leq n; j++)$ **do**
        Calculate excess power given to j^{th} FCP according to Eq. (4.15)
        Manage the charging rate of j^{th} PHEV according to Eq. (4.16)
    end for
end for
▷ **Instantaneous load change**
When a PHEV joins or leaves i^{th} CS at t^k time instant
Update the value of $P^i_{CS}(t)$ at t^k
if $(X(t) > G(t))$ **then**
    Calculate $P^{draw}_{b_{ji}}(t^k)$ for each SCP according to Eq. (4.11) and manage its C^{new}_{ji} according to Eq. (4.12)
else
    Calculate $P^{excess}_{b_{ji}}(t^k)$ for each FCP according to Eq. (4.15) and manage its C^{new}_{ji} according to Eq. (4.16)
end if

4.4.2 Valley filling using PHEVs

In addition to peak shaving, PHEVs (labeled as FCPs) are used to fill the valley in demand at smart grid as given in algorithm 5. Valley in demand implies that smart grid has excess power (say $P^{exc}(t)$) which needs to be distributed to all the CSs. This power is distributed to each CS at t^{th} time slot according to the equation given below.

$$P_{CS_i}^{excess}(t) = \frac{P_{CS}^{i}(t)}{P_{CS}(t)} P^{exc}(t) \tag{4.13}$$

such that

$$\sum_{i=1}^{m} P_{CS_i}^{excess}(t) = P^{exc}(t) \tag{4.14}$$

where, $P_{CS_i}^{excess}(t)$ is the excess power distributed to i^{th} CS. This power is further distributed amongst different FCPs present at i^{th} CS according to the following equation.

$$P_{b_{ji}}^{excess}(t) = \frac{P_{b_j}^{i}(t)}{P_{CS}^{i}(t)} P_{CS_i}^{excess}(t) \tag{4.15}$$

where, $P_{b_{ji}}^{excess}(t)$ is the excess power given to j^{th} FCP's battery at i^{th} CS. The charging rate of this PHEV's batteries is altered according to the equation given below.

$$C_{ji}^{new} = C_{ji}\left(1 + \frac{P_{b_{ji}}^{excess}(t)}{P_{b_j}^{i}(t)}\right) \tag{4.16}$$

such that

$$\begin{cases} 1 < C_{ji}^{new} \leq 2; \\ SoC_{cur} \leq SoC_{des} \end{cases}$$

The instantaneous load change at the CS during valley filling is handled as follows. When a new PHEV joins or leaves the CS at any time instant, then its power is compensated locally by re-calculating the values of $P_{b_{ji}}^{excess}(t)$ and C_{ji}^{new} in a similar manner as discussed in peak shaving.

4.5 Performance Evaluation of Demand Response Management Scheme

The results obtained after performing simulation of the proposed demand response management scheme are summarized in this section. A typical distribution network is considered for simulation purpose having the following parameters. For the sake of simplicity, power generation capacity in this network is considered to be 1 MVA with power factor taken as unity (i.e., $\forall\, t,\ G(t) = 1$ MW). In the considered network, the scaled-down residential load curve of actual PJM data of sub-zone PLCO on a typical day of summer has been considered [139]. This demand curve is assumed to comprise the load requirements of 200 smart homes having various smart devices as given in [129]. As the dataset given in [129] comprises of hourly load demand of the devices, the time slots of one hour are considered in the proposed scheme (however it can handle time slots of lesser duration as well). In addition to it, 100 PHEVs with battery capacity of 12-16 kWh are assumed in the network which can charge at any of the 10 considered CSs using the strategy similar to [132].

The working of proposed scheme is as follows. Initially, the consumption data of residential sector and PHEVs for every time slot is collected at the utility server as shown in Fig. 4.5. The aggregated demand of these loads at smart grid is then calculated at the server which is illustrated in Fig. 4.6 along with its power generation capacity.

It can be inferred from this figure that smart grid faces peak and valley in demand during different times of the day with maximum peak during 1300 to 1500 hours. During these time slots, the demand response of different loads is managed as discussed in Section 4.2 which alters the load profile of grid as shown in Fig. 4.6.

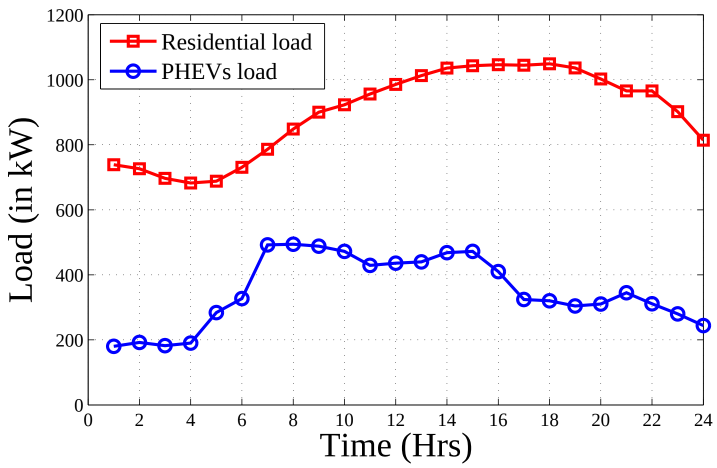


Figure 4.5: Demand profile of different loads.

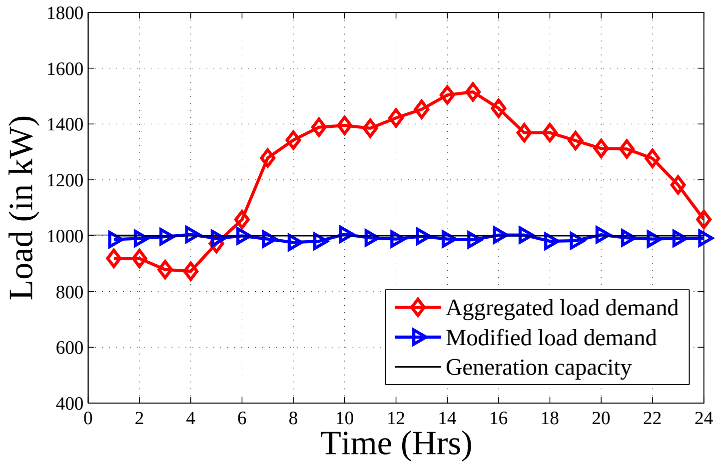


Figure 4.6: Demand curve at the smart grid.

For the residential loads, the SVM classifier identifies the smart homes with excess load consumption as discussed in Section 4.3.1. Similarly, for the PHEVs placed at various CSs, the SVM classifier labels PHEVs into FCPs/SCPs or NCPs as described in Section 4.4. For simulation purposes, the feature set of smart homes is acquired from [129], [130], [131]; whereas the feature set for PHEVs is generated statistically. Once the feature set for both smart homes and PHEVs is collected, these features are normalized in the range [0,1] using Eq. (4.1) and Gaussian RBF kernel is used for classification of data. The grid search and cross validation methods are implemented in LIBSVM [125] to compute the optimized values of C and γ. Using these methods, the values of C and γ comes out be 2 and 1 respectively (for smart homes) and 1.94 and 0.92 respectively (for PHEVs). The complete dataset of smart homes and PHEVs is divided into 70-30 ratio such that 70% of values are used for training and 30% values are used for testing. The accuracy of SVM classifier for classification of smart homes and PHEVs on this division is shown in Fig. 4.7. It can be inferred from this figure that the considered 10-fold cross validation testing accuracy of this classifier is very high (88.5% in case of smart homes and 94% in case of PHEVs).

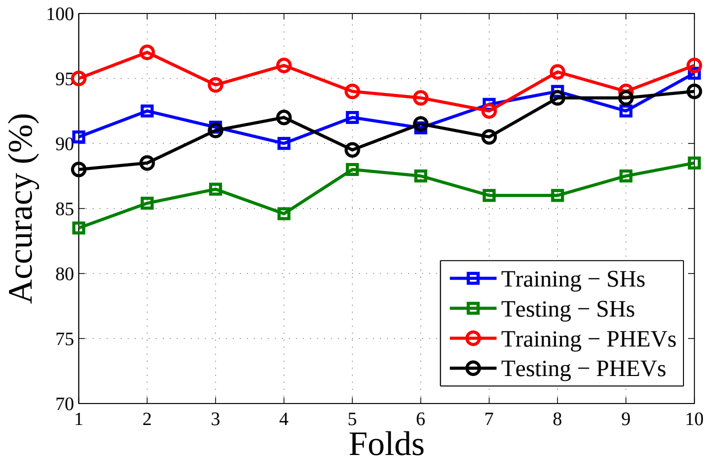


Figure 4.7: Cross validation accuracy of SVM.

For classification of smart homes and PHEVs, the computing time required by the proposed scheme on the basis of number of requests is shown in Fig. 4.8. This computational analysis has been carried out by using Intel Core i3-3110M CPU @ 2.40 GHz with 4GB of primary memory. It is to be noted that the communication delay for gathering the data is not considered while depicting these results. Based on SVM classification, the demand response of smart homes and PHEVs is managed; the results of which are presented in the subsequent sections.

4.5.1 Residential load management

The demand of smart homes, classified as using excess load consumption, is managed by re-scheduling the load of devices according to algorithm 3. For one such smart home, the initial load consumption of various devices on a typical day of summer is

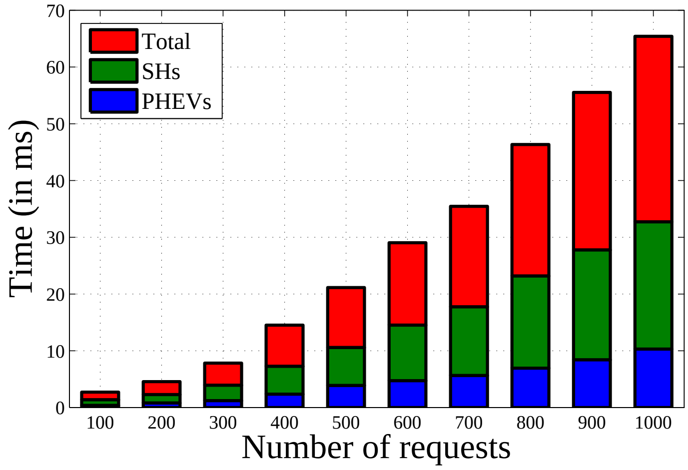


Figure 4.8: Time taken for SVM classification.

shown in Fig. 4.9. The loads in this smart home are re-scheduled into various time

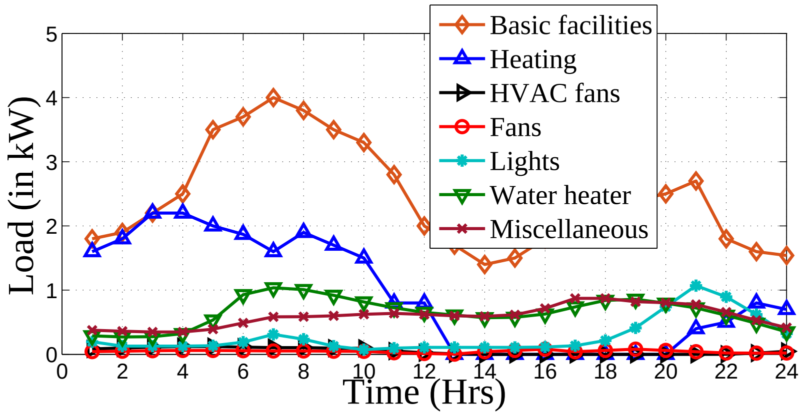


Figure 4.9: Load of devices in a smart home.

slots as shown in Fig. 4.10. In this figure, it can be seen that 1.5 kW of load is shed at 0800 hours to reduce the load during peak time. It can also be observed from Fig. 4.10 that 0.4 kW of load is added at 1300 hours while the smart grid is still in peak load condition. This is because water heater (which was switched OFF at 1200 hours) has to be switched back ON at 1300 hours as specified in the rule-base given in Table 4.1. Using such rules, the loads in different smart homes are rescheduled which leads to the change in overall demand of residential load as depicted in Fig. 4.11. However, managing the demand response of smart homes alone is not able to maintain the overall load stability at smart grid. This is evident from Figs. 4.6 and 4.11 where the aggregated load at 0900 hours is 1388.3 kW which implies that 388.3

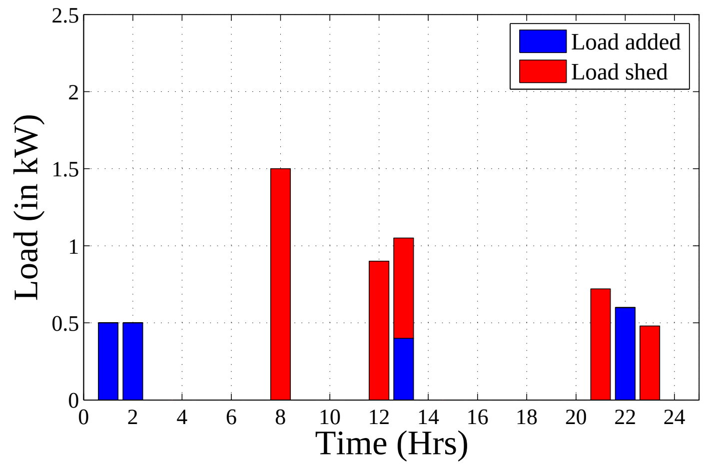


Figure 4.10: Rescheduling of load in a smart home.

kW of load needs to be shed. In this scenario, the residential load accounts for only 290 kW which still leaves the smart grid in peak load condition. To compensate for the remaining power, the charging rates of PHEVs are altered.

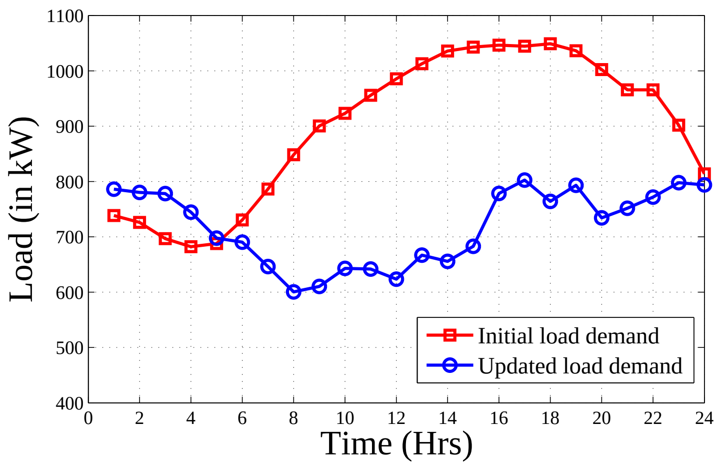


Figure 4.11: Demand curve of residential load.

4.5.2 Managing PHEVs at CSs

The charging rates of PHEVs (identified by SVM) present at various CSs are regulated according to algorithm 5 as described in Section 4.4. Fig. 4.12 shows the number of PHEVs present at different CSs during the time slot of 0900 to 1000 hours. The alteration in the charging rates of the PHEVs, labeled as SCPs and FCPs dur-

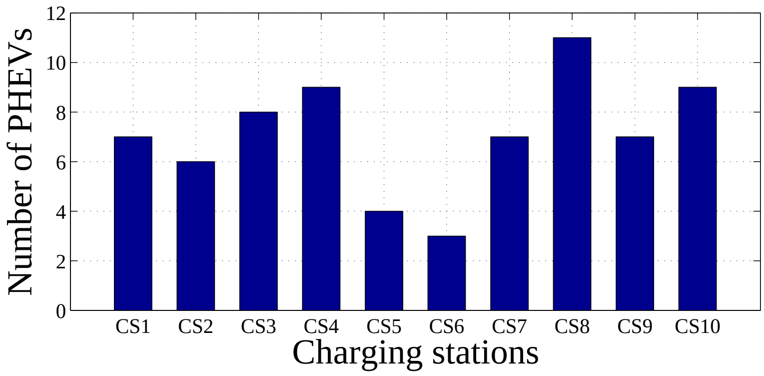


Figure 4.12: PHEVs present at different CSs.

ing peak and off-peak hours respectively, modifies the overall demand requirement of the CSs which is depicted in Fig. 4.13.

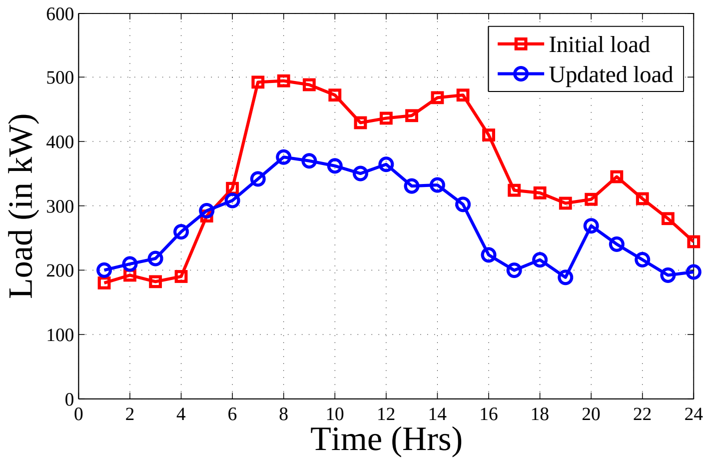


Figure 4.13: Load demand of CSs.

The variation in charging rates of single PHEV in case of slow charging (during peak hours) and fast charging (during off-peak hours) cases for four time slots is depicted in Fig. 4.14; while in the normal charging, the charging rate is always set to 1. In this figure, the time slots during peak time are considered from 0700 to 1000 hours while the off-peak time slots are considered from 0200 to 0500 hours. During peak shaving, i.e., at 0900 hours, 10.1 kW power is required from the considered CS (as per Eq. (4.9)) and the charging rates of SCPs at this CS are managed using Eq. (4.12). During the valley filling, i.e., at 0500 hours, 2.8 kW of excess power is given to this CS (according to Eq. (4.13)) and charging rates of its FCPs are regulated

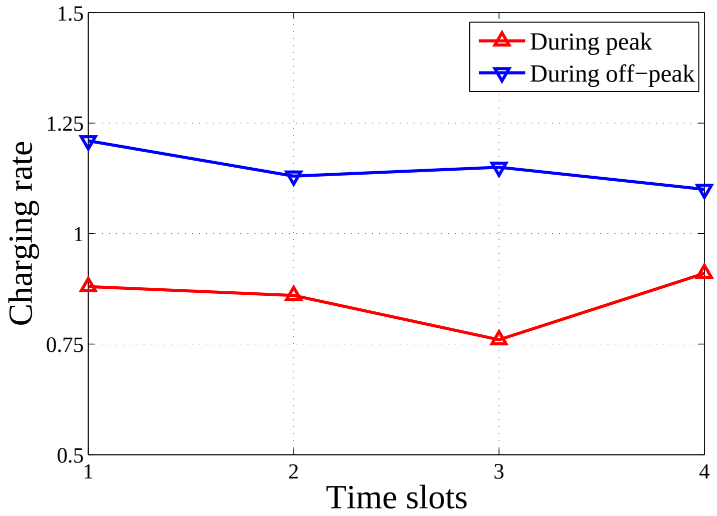


Figure 4.14: Variation in PHEV's charging rate.

using Eq. (4.16). The change in SoC level of a PHEV used as FCP, SCP and NCP corresponding to the above scenario is given in Fig. 4.15. The battery energy rating of this PHEV is taken as 12 kWh with 50% initial SoC level.

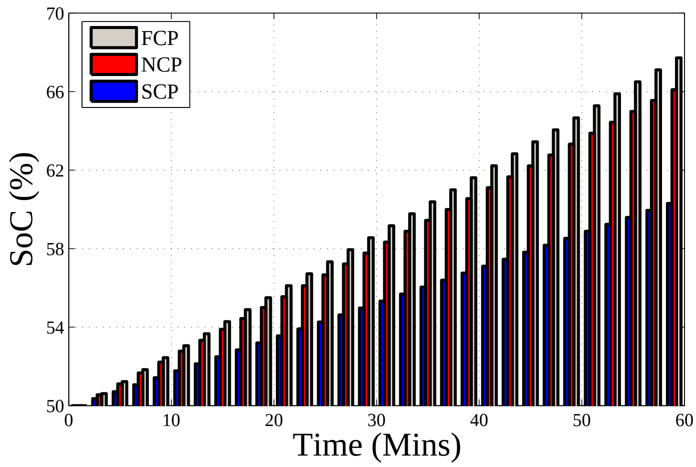


Figure 4.15: Variation in PHEV's SoC level.

4.6 Summary

This chapter presented a novel scheme to handle the demand response of smart homes and PHEVs in the energy network to maintain the load stability in smart grid at all the times. To achieve this task, data analytics is applied to manage the demand response of these loads. A binary-class SVM classifier has been employed to identify the residential users using excess electricity consumption; and the load of such smart homes is then curtailed using a load balancing algorithm. If the residential loads are not sufficient to reduce the demand and supply gap, then the PHEVs are used as variable loads to manage the load stability. For this purpose, the SVM classifier discovers the PHEVs whose charging rates can be regulated. The load profile of smart grid is then managed by altering the charging rates of such PHEVs according to the grid requirements. The proposed scheme harnesses the existing capabilities of smart grid to handle the demand response and also alleviates the need for utilities to build new infrastructure in order to manage their power demands. The simulation results prove that the proposed scheme is effective in maintaining the overall load profile of smart grid. Moreover, the classification time taken by the SVM classifier is very less and its accuracy is very high; thereby making the overall scheme very fast and reliable.

Chapter 5

Peak Load Reduction

The rapid increase in load demand of various sectors such as-residential, commercial, industrial and transportation has escalated the burden on the grids. Among all these sectors, the residential sector is the biggest consumer which constitutes approximately 38% of the total electricity consumption in the U.S. and is further expected to grow by 0.3% annually [1]. To tackle this issue, a novel data analytical demand response (DADR) management scheme for residential load is proposed in this chapter with an aim to reduce the peak load demand in smart grid.[1] The proposed scheme is primarily based on the analysis of consumers' consumption data gathered from smart homes for which factors such as-appliance adjustment factor, appliance priority index, appliance curtailment priority, etc. have been designed. Based on these factors, different algorithms with respect to the consumer's and utility's perspective have been proposed to take demand response decisions in the peak load scenario. In addition to it, an incentive scheme has also been presented to increase the consumers' participation in the proposed scheme. The proposed scheme is tested on the dataset gathered from PJM and Open Energy Information. The results obtained show that it efficiently reduces the peak load at the grid by a great extent.

[1]The contents of this chapter are partly published in: A. Jindal, M. Singh, and N. Kumar, "Consumption-Aware Data Analytical Demand Response Scheme for Peak Load Reduction in Smart Grid," *IEEE Transactions on Industrial Electronics,* vol. 65, no. 11, pp. 8993-9004, 2018.

5.1 Contributions

The major contributions of this chapter are:

1. Various factors such as-appliance adjustment factor, appliance priority index, appliance curtailment priority, etc. are designed by analyzing the consumption data of smart homes.

2. Different algorithms from consumer's as well as utility's perspective have been proposed on the basis of these factors which are responsible for taking demand response decisions in smart homes so as to reduce the peak load at smart grid. Moreover, instantaneous load changes in smart homes are also managed in real-time to maintain the load profile of the grid.

3. A consumer incentive scheme, based on user-specified curtailment factor and the actual load curtailed in smart homes, is also presented in this chapter. This scheme compensates for the user discomfort and also increases the consumer participation in the demand response management process.

5.2 Proposed Scheme

This section presents the detailed description of the proposed scheme used for the demand response management of smart homes so that the load on smart grid during peak hours can be reduced in real-time at the start of each time-slot. The pricing policy used in the proposed scheme is considered as fixed to ensure that there is no rebound peak at the grid [140]. To apply the proposed scheme in real electricity market, it is necessary to communicate the energy data of smart homes to the utility. The communication scenario between smart homes and utility server to gather this data is shown in Fig. 5.1. The security and privacy of this data can be ensured by implementing some lightweight cryptographic security mechanisms which do not affect the overall system performance [141]. As shown in this figure, the complete data gathering process has been divided into three layers. In layer 1, the devices in smart homes communicate their usage data to the local controller present in the respective homes. Layer 2 helps in transferring the data collected at the local controller to the smart grid server in layer 3 with the help of access points (APs). Once the data is gathered at the smart grid server, the data analytics is performed to take demand response decisions which are then relayed back to the local home controllers to control the devices in the respective homes.

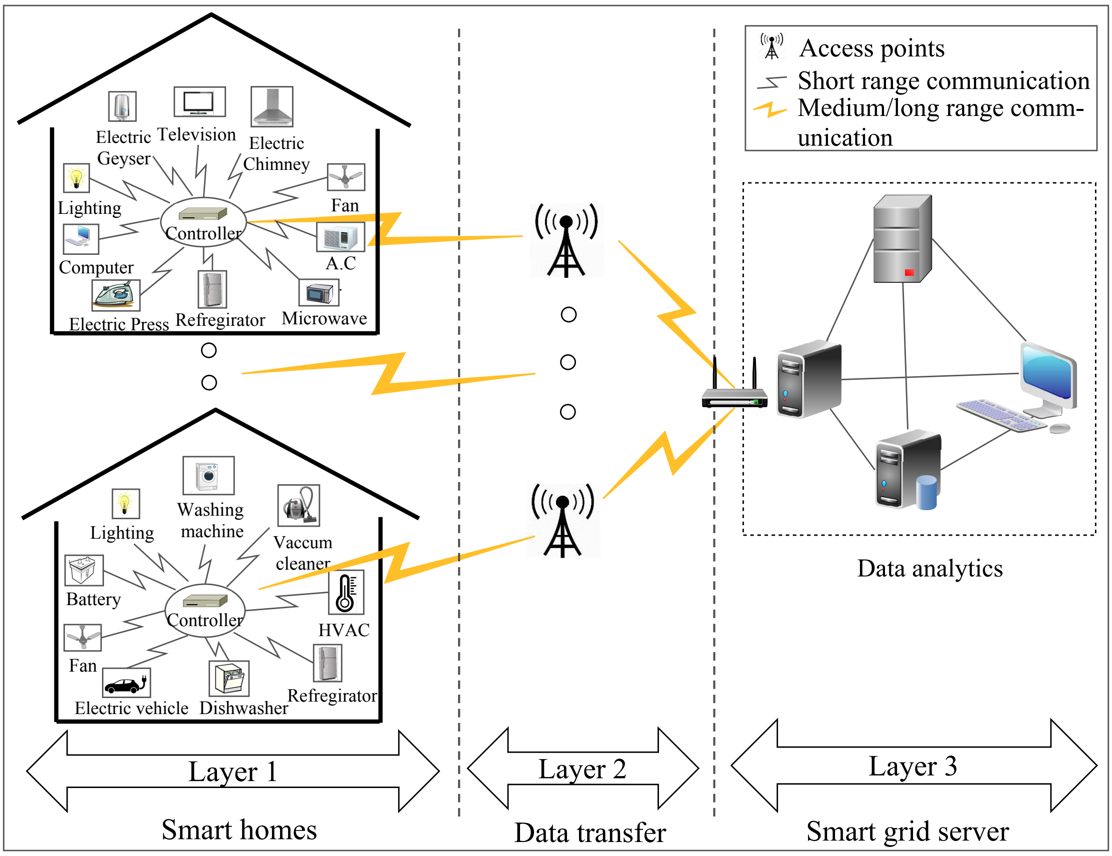


Figure 5.1: Communication of smart homes with the smart grid server.

The complete day is divided into numerous time-slots of equal time intervals where the peak load is managed as follows. Let the generation capacity of grid at the start of t^{th} time-slot be $G(t)$ (in kW) (considering installed capacity as well as the renewable energy generation), total number of smart homes be n, and power consumption requirement in i^{th} home at t^{th} time-slot be $H_u(i,t)$ (in kW). The load at the grid needs to be managed if the following condition exists.

$$\sum_{i=1}^{n} H_u(i,t) > G(t) \tag{5.1}$$

For such a case, the total energy deficit ($E_{def}^{grid}(t)$) at the grid in order to reduce the load demand is,

$$E_{def}^{grid}(t) = \left(\sum_{i=1}^{n} H_u(i,t) - G(t) \right) \times \frac{\Delta t}{60} \tag{5.2}$$

where, Δt is the time duration of t^{th} time-slot.

To fill the deficit specified in (5.2), the load demand in the smart homes is curtailed on the basis of data collected from smart homes. For this purpose, the proposed scheme uses ZigBee and IEEE 802.15.4a/c/f technologies to communicate the data from various appliances to local controller; while uses WiFi, WiMAX and long term evolution (LTE) technologies to communicate data from local controller to the utility server [96, 142]. All these technologies have very high data rates varying from 250 Kbps to 1 Gbps and hence the communication delays in collection of data are negligible. However, the speed and memory concerns for storage and analysis of collected data may arise. To cater these issues, the utility server can be placed at cloud or at a centralized place with abundant resources. However, as a centralized system, it is vulnerable to single point of failure. In order to address this concern, utility can employ strategies such as-checkpoint algorithm [143] to ensure safety and fast recovery of the system. After collection of data at the server, different factors are computed based on which the demand response decisions are made. These decisions are communicated back to the individual controllers of smart homes for automated switching of appliances [97, 144]. Apart from it, an incentive is provided to the users in lieu of their participation in the proposed scheme. The flowchart of the complete scheme has been shown in Fig. 5.2 which can be understood with the help of following sections.

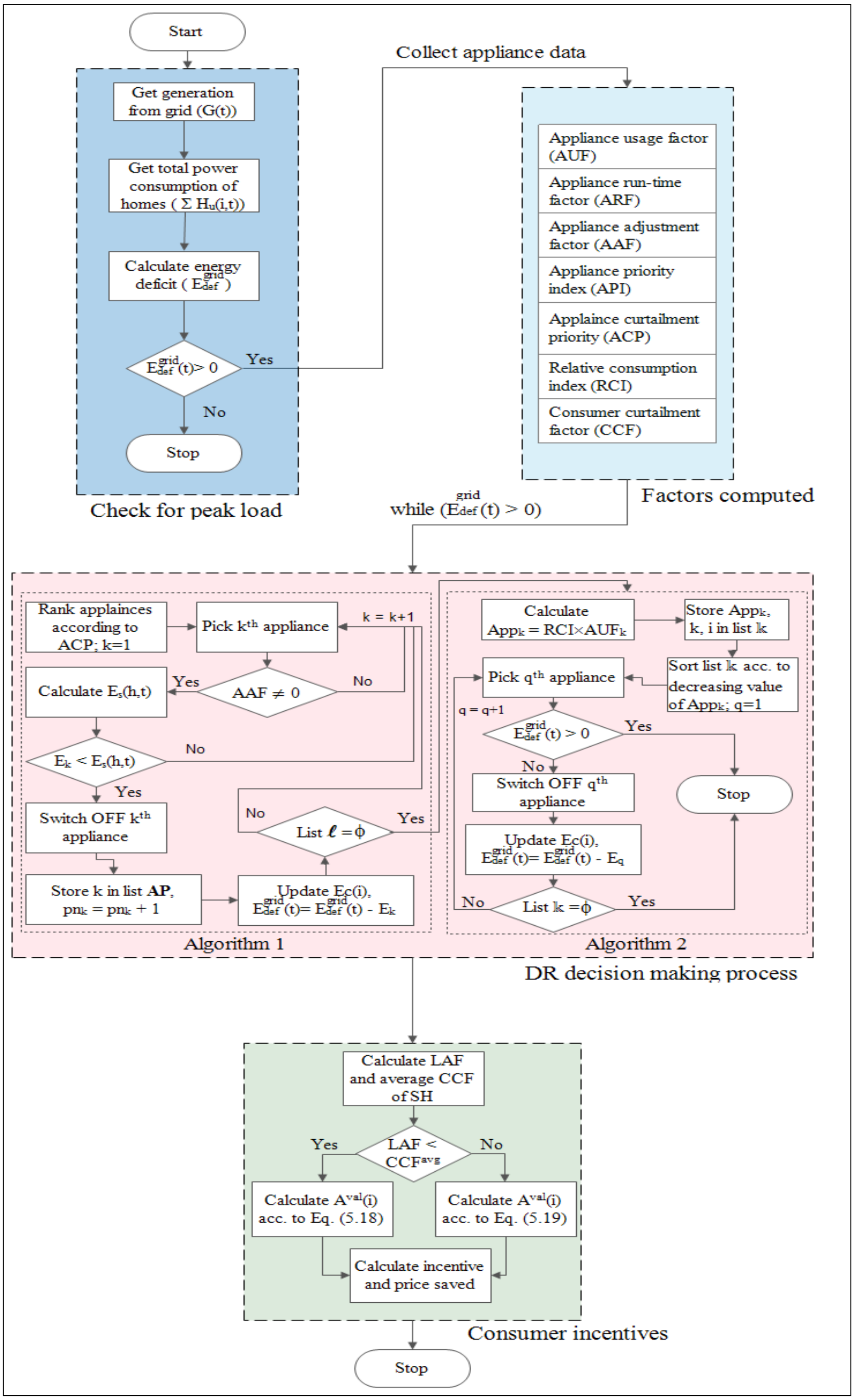


Figure 5.2: Flowchart of the proposed DADR scheme.

5.3 Factors used in Demand Response Decision Making

This section explains how various factors that directly influence the demand response decision making process are computed. These factors are primarily dependent on the data gathered from the smart homes and can be easily calculated by analyzing this data for each time-slot. Each factor has its own significance (as discussed below) which makes it a very important entity in the overall demand response management.

5.3.1 Appliance usage factor (AUF)

The AUF gives the ratio of the power utilized by an appliance with respect to the total power consumption in a home. It is represented as,

$$AUF_j(i,t) = \frac{P_j^{rated}}{H_u(i,t)} \tag{5.3}$$

where, $AUF_j(i,t)$ represents the AUF of j^{th} appliance for i^{th} home at t^{th} time-slot and P_j^{rated} is the rated power of j^{th} appliance (in kW). The calculation of AUF is important to identify the appliances with maximum load requirement in a smart home. The appliance with largest value of AUF suggests that it consumes maximum energy.

5.3.2 Appliance run-time factor (ARF)

ARF is the factor which depicts the ratio of time for which a particular appliance was in running state (or ON) during the previous time-slot. For example, if an appliance 'A' was ON for 12 minutes in a time-slot of 15 minutes, then, the ARF for 'A' would be 0.8.

The ARF for j^{th} appliance in i^{th} smart home is calculated with the help of data gathered about its running status in the previous time-slot as given below.

$$ARF_j(i, t-1) = \frac{T_j^{run}(t-1)}{\Delta t} \tag{5.4}$$

where, $T_j^{run}(t-1)$ is the duration for which the j^{th} appliance was ON in $(t-1)^{th}$ time-slot. The ARF is used to depict the amount of work an appliance has done in the previous time-slot based on which the adjustment decisions are taken as discussed in next segment. The value of ARF closer to 1 indicates that the appliance remained ON for most of the duration in the previous time-slot.

5.3.3 Appliance adjustment factor (AAF)

The factor, AAF, represents that whether the appliance can be adjusted (i.e., considered for curtailment) by the proposed scheme. Its value is set to either 0 or 1 depending upon the following condition and only the appliances for which $AAF = 1$ will be adjusted in the proposed scheme.

$$AAF_j(i,t) = \begin{cases} 0 & if\ ARF_j(i, t-1) > \tau\ \&\&\ S_j(i) = 1\ \&\&\ j \in D; \\ & j \in E \\ 1 & otherwise \end{cases} \tag{5.5}$$

where, $AAF_j(i,t)$ is the adjustment factor for j^{th} device of i^{th} smart home, τ is the predefined threshold value and $S_j(i)$ represents the present state of the j^{th} device whose value is calculated as,

$$S_j(i) = \begin{cases} 1 & if\ \ device\ 'j'\ is\ ON \\ 0 & otherwise \end{cases} \tag{5.6}$$

Apart from these values, there are two lists D and E which are also used in calculating the value of AAF. These lists store the information about different type of appliances. The list E contains all emergency appliances which are kept out of

DADR scheme like lights, fans, bulbs and healthcare devices. So, the AAF for all such appliances is set to 0. The list D contains various appliances which include– washing machine, electric iron and all the kitchen appliances like microwave, electric cooker and oven.

The importance of AAF is that the devices in list D should not be adjusted after completing some part of their cycle time (specified by τ), otherwise the job at hand can be hampered. For example, if the washing machine is stopped after a while, then, the clothes would remain wet and may leave stains. Similarly, if someone is making an egg pie and suddenly the microwave is switched OFF, then the taste of egg pie would be bitter as it requires constant heating while baking. So, such devices should not be adjusted if they have been running for some time in the previous time-slot and are still ON. The value of τ is taken to be 0.5 in the proposed scheme; this is because the switching OFF appliances during initial few minutes would not hamper the whole process (This value can be changed anytime by the utility on the request of the users). Moreover, AAF is particularly important to keep the emergency appliances out of the demand response management process.

5.3.4 Appliance priority index (API)

The API is the most important index and is calculated by analyzing the appliance data collected from various smart homes. It is used to measure the priority of one appliance over the others which is utilized while curtailing the load demand. It also takes user's comfort into account as the usage of this appliance in other homes with similar load conditions is analyzed while calculating this priority. The motivation for considering other homes with similar load conditions for calculating API is that they exhibit same kind of response in terms of their consumption pattern. Thus, they act as a better indicator for calculating appliance priority in a smart home rather than considering all the smart homes. These homes are discovered by consumer profiling and forming clusters such that each cluster portrays similar behavior of its participants. In this regard, a popular clustering technique known as the standard k-means clustering has been used in the proposed scheme [44,45]. The steps involved in k-means clustering are given below.

1. Assign each consumer randomly to any cluster c ($c = 1, 2, \ldots, C$)

2. Compute the centroid for each cluster as:

$$\mu_c = \frac{1}{N_c} \sum_{x_i \in C_c} x_i \tag{5.7}$$

   where, μ_c is the centroid of c^{th} cluster, x_i is the feature value of i^{th} consumer, C_c is the set of consumers in cluster c and N_c is the number of consumers in C_c.

3. Reassign each consumer to the cluster for which distance of feature value and new centroid ($d(x, \mu)$) comes out to be minimum.

$$d(x, \mu) = \sqrt{\frac{1}{S} \sum_{s=1}^{S} (x_s - \mu_s)^2} \tag{5.8}$$

   where, x and μ represents the feature vector of all consumers and cluster centroids respectively, and S is the size of feature vector.

4. Repeat steps 2 and 3 until there is no significant change in the centroid of clusters (i.e., clusters become stable).

The clusters, on the basis of H_u, are formed for each time-slot using the above mentioned steps to take into account the dynamic nature of appliances. Once the clusters becomes stable, the API of an appliance j ($API_j(i, t)$) is computed as given below.

$$API_j(i, t) = \vartheta \left(\frac{\sum_{k=1}^{m} \theta_{jk}}{m \times n} \right); k \in C_d \tag{5.9}$$

where, n is the total number of smart homes connected to smart grid, m is the number of smart homes in cluster C_d (C_d is the cluster which exhibits same behavior as i^{th} smart home in terms of H_u and is determined using Eq. (5.8)) and ϑ is the scaling constant. The value of θ_{jk} represents the present working status of appliance $'j'$ in k^{th} smart home which is computed by,

$$\theta_{jk} = \begin{cases} 1 & if\ 'j'\ is\ ON\ in\ k^{th}\ smart\ home \\ 0 & otherwise \end{cases} \tag{5.10}$$

The numerator on right hand side of Eq. (5.9) calculates the total number of smart homes from cluster C_d in which the j^{th} appliance is ON. This value is then divided by the total number of smart homes in cluster C_d (m) and total number of smart homes (n) to calculate the priority of appliance j in i^{th} smart home. This calculation is based on the fact that more frequently used appliances in similar smart homes should have more priority over the less utilized appliances. Thus, the higher value of API for an appliance signifies that the considered appliance has more priority in a smart home as compared to other appliances. To understand the concept of API more clearly, a small case study is presented as follows.

Case Study: Let us say that there are 200 smart homes in total (i.e., $n = 200$) and the API of A.C. for one smart home needs to be calculated. For this smart home, the value of H_u is assumed to be 3 kWh. To calculate the value of API, initially these smart homes are divided into various clusters using k-means clustering algorithm. Fig. 5.3 illustrates the set of centroid values for different number of clusters after stabilization. It can be inferred from this figure that for $C = 4$, the k-means clustering algorithm is able to richly depict the load profiles of various smart homes. Thus, four clusters $\{C_1, C_2, C_3, C_4\}$ are chosen to calculate the value of API which comprise of {39, 56, 62, 43} smart homes, respectively. Suppose, these clusters have $\{\mu_1, \mu_2, \mu_3, \mu_4\}$ as {1.85, 2.95, 3.74, 4.62} after stabilization in the considered time-slot. As the value of μ_2 is closest to the H_u of given smart home, the total number of smart homes (with status of A.C. as ON) are extracted from C_2 using Eq. (5.10). Lets say this number comes out to be 31; using this value in Eq. (5.9), the API for A.C. of the considered smart home is computed as 0.138 (considering value of ϑ to be 50).

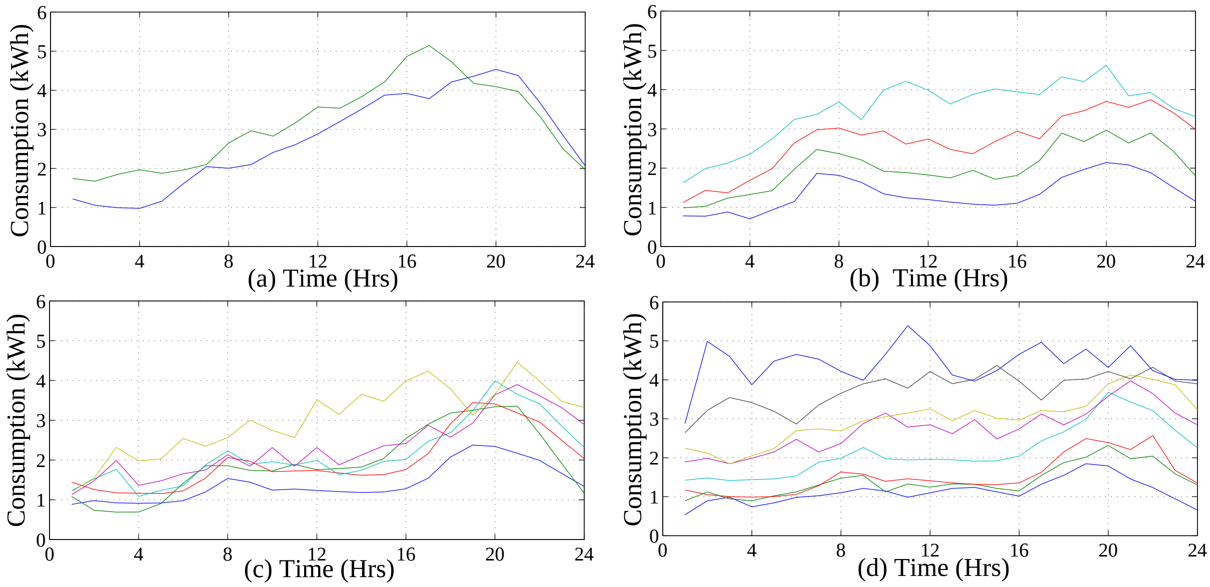


Figure 5.3: Cluster centroids with respect to number of clusters (a) C=2 (b) C=4 (c) C=6 (d) C=8.

5.3.5 Appliance curtailment priority (ACP)

The importance of factor 'ACP' is that it is used to increase the priority of an appliance in a smart home which is frequently switched ON by the user (after being switched OFF by the utility). Thus, the higher value of ACP means that it is of high priority to the user and should not be curtailed. This factor is calculated as given below.

$$ACP_j(i,t) = \begin{cases} API_j(i,t) & if\ pn_j = 0 \\ API_j(i,t)(1 + \gamma \times pn_j^2) & otherwise \end{cases} \tag{5.11}$$

where, $ACP_j(i,t)$ is the value of ACP of j^{th} appliance in i^{th} smart home at t^{th} time-slot, γ is the scaling constant and pn_j is the number of times the appliance

j has been switched OFF by the utility on a given day for i^{th} smart home. The value of $API_j(i,t)$ in the above equation is computed using Eq. (5.9) and value of pn_j is extracted from list AP (created using algorithm 6). The list AP contains pn values of all the appliances for all the smart homes that were curtailed by the utility. The variation in values of ACP of an appliance for different values of pn and API is shown in Fig. 5.4.

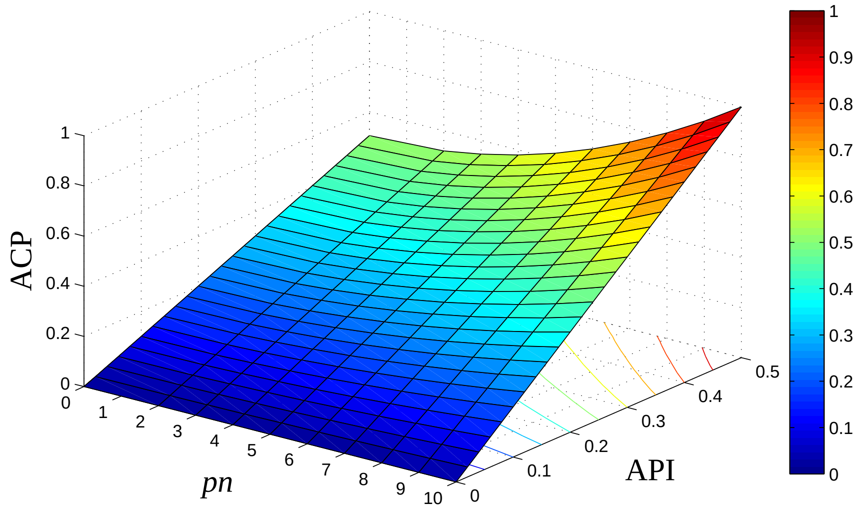


Figure 5.4: Variation in values of ACP.

5.3.6 Relative consumption index (RCI)

The importance of RCI is that it helps in identifying the homes with high load consumption. The value of RCI for i^{th} smart home for t^{th} time-slot is given by,

$$RCI(i,t) = \frac{H_u(i,t)}{\max_{(k \in C_d)}\{H_u(k,t)\}} \tag{5.12}$$

The above equation would compute the relative consumption of i^{th} smart home with respect to the smart home having maximum consumption in the same cluster, C_d. The higher value of RCI implies that the i^{th} smart home is consuming more energy as compared to the other smart homes having lower values of RCI.

5.3.7 Consumer curtailment factor (CCF)

CCF is the only factor that the consumer needs to specify in the proposed DADR approach. It is the maximum desired limit on the load that can be curtailed by the utility at any time. If the value of CCF is specified to be 1%, then, not more than 1% of the total load would be curtailed. For example, if the cumulative load in a smart home is 10 kWh after a time-instant and specified CCF is 2%, then, the load profile at that time-instant can be adjusted by 0.2 kWh. The consumers can increase or decrease their CCF value to control their participation in demand response management process. An instance of specified values of CCF by the consumers of four smart homes for a given day has been depicted in Fig. 5.5. It is to be noted that the value of CCF would be reset to 1% at the start of next day at a particular time-slot (considered 06:00 AM in the proposed scheme), if not changed by the user.

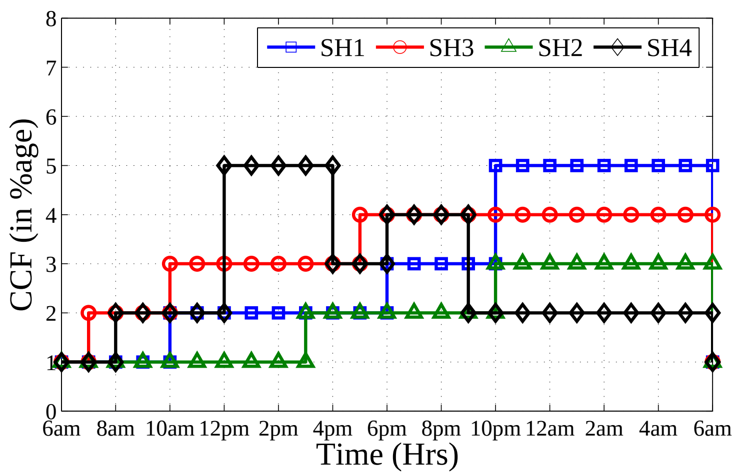


Figure 5.5: Specified CCF for four smart homes.

The CCF is also one of the measures used to compute consumer satisfiability and comfort as the consumer would specify this according to his suitability. The consumer is assumed to be satisfied if load curtailed is below or equal to his specified CCF value. The energy that can be saved by load curtailment during t^{th} time-slot

$(E_s(i,t))$ is based on this CCF as given below.

$$E_s(i,t) = CCF(i,t)\left(\sum_{k=1}^{t} H_u(i,k) \times \frac{\Delta t}{60}\right) - E_c(i) \tag{5.13}$$

where, $CCF(i,t)$ is the CCF specified by the user in i^{th} smart home at t^{th} time-slot. In the above equation, $E_c(i)$ is the cumulative value of energy that has been saved through actual load curtailment for i^{th} smart home (determined by Eq. (5.14)). The lower limit of $E_s(i,t)$ is set to 0 so as to avoid any negative values.

$$E_c(i) = \begin{cases} 0 & if\ t = 1 \\ E_c(i) + E_j & otherwise \end{cases} \tag{5.14}$$

where, E_j is the energy of j^{th} curtailed appliance for Δt duration and it is calculated as,

$$E_j = P_j^{rated} \times \frac{\Delta t}{60} \tag{5.15}$$

Eq. (5.14) specifies that the value of $E_c(i)$ is set to 0 at the start of the day. For other instances, the energy saved by switching OFF an appliance (say j) is added to the value of $E_c(i)$.

5.4 Taking Demand Response Decisions

The factors defined in previous section are used to take demand response decisions with the help of two proposed algorithms. First algorithm (named as consumer centric (CC)-DADR algorithm) is from consumer's perspective and reduces the peak load by taking user's comfort into account. The load curtailment in this algorithm is dependent on the user specified value of CCF and will not exceed this value. As a result, this algorithm may not completely reduce the peak load, but it ensures that the load curtailment do not hamper the comfort of the user. Second algorithm (named as utility centric (UC)-DADR algorithm) is from grid's perspective which ensures that the peak in its load demand is further reduced; however, in doing so, user's comfort may get violated. Thus, in order to compensate for this violation, extra incentive will be given to the consumers. The utility may choose to follow just CC-DADR algorithm or both CC-DADR and UC-DADR algorithms depending upon its constraints. Using these algorithms, the load profile of smart grid is maintained by curtailing the load demand from smart homes as discussed below.

5.4.1 CC-DADR algorithm

Initially, algorithm 6 is used to reduce the load in smart homes according to the consumer-specified CCF value. The first step in this algorithm is to calculate the API and ACP of all the appliances in all the smart homes. These appliances are then stored in list ℓ according to increasing values of their ACP. The significance of list ℓ

Algorithm 6 Consumer centric (CC)-DADR algorithm

for $(i = 1; i \leq s; i++)$ **do** ▷ s⟻ Total number of smart homes
    $\forall\ j$, calculate API using Eq. (5.9) ▷ j⟻ Appliances in a smart home
    $\forall\ j$, calculate ACP using Eq. (5.11)
end for
Sort $ACP_j(i,t)$ according to ascending order and store in list ℓ
while $(E_{def}^{grid}(t) > 0 \mid\mid \ell == \emptyset)$ **do**
    for $(k = 1; k \leq n; k++)$ **do** ▷ n⟻ Total appliances in ℓ
        Extract associated home (in h)
        Calculate AAF of k^{th} device using Eq. (5.5)
        if $(AAF_k(h,t) \neq 0)$ **then**
            Get CCF for home h
            Calculate $E_s(h,t)$ using Eq. (5.13)
            if $(E_k < E_s(h,t))$ **then**
                Switch OFF appliance k
                Store appliance k in list AP; $pn_k = pn_k + 1$
                Communicate this decision to the local controller of h
                Update $E_c(i)$ using Eq. (5.14)
                Update $E_{def}^{grid}(t) = E_{def}^{grid}(t) - E_k$
            end if
        end if
    end for
end while

is that the top appliance in this list depicts that it has been least used in other smart homes and thus, its priority is set to the lowest. For all these appliances (indexed by k), the value of AAF is computed and the appliances for which $AAF \neq 0$ are curtailed in the following way. The value of $E_s(i,t)$ for the corresponding smart home is calculated (which is based on CCF) and if $E_k < E_s(i,t)$, then, appliance k is switched OFF. The value of pn for k^{th} appliance is increased and its information is stored in the list AP. The values of $E_c(i)$ and $E_{def}^{grid}(t)$ are updated accordingly and the complete process is repeated until the list ℓ is exhausted or $E_{def}^{grid}(t) < 0$. In this way, algorithm 6 curtails the load in homes according to user-specified value of CCF.

5.4.2 UC-DADR algorithm

While complying with CCF of every smart home in algorithm 6, it may be the case that the load on grid is still more than its generation capacity. In such a scenario, the algorithm 7 is used to stabilize the load profile of the grid for managing the utility constraints. The working of this algorithm is as follows. Initially, the value of RCI for every smart home and AUF for every appliance is computed. To switch OFF the appliances, the value of a new variable App_k is calculated and this information is stored in a list $\Bbbk$. The appliances which have higher values of App_k are switched OFF and the values of $E_{def}^{grid}(t)$ and $E_c(i)$ are updated accordingly. The complete process is repeated till $E_{def}^{grid}(t) > 0$ or the appliances are available in list $\Bbbk$. The significance of algorithm 7 is that it switches OFF the devices which consumes more energy in smart homes and put more burden on the grid. In this way, the peak load in smart grid is reduced using the proposed scheme.

To manage the load in the off-peak hours, the loads stored in list $\Bbbk$ and AP respectively are switched ON until $E_{def}^{grid}(t) \leq 0$. However, as the main focus of this research is to reduce the peak load, the load management in off-peak hours is not discussed in detail.

Algorithm 7 Utility centric (UC)-DADR algorithm

for $(i = 1; i \leq s; i++)$ **do** ▷ s⟵Total number of smart homes
    Calculate RCI using Eq. (5.12)
    for $(k = 1; k \leq m; k++)$ **do** ▷ m⟵Total appliance in a smart home
        Calculate AUF using Eq. (5.3)
        Calculate $App_k(i,t) = RCI(i,t) \times AUF_k(i,t)$
        Store value of $App_k(i,t)$, k and i in list $\mathbb{k}$
    end for
end for
Sort list $\mathbb{k}$ in descending order of value $App_k(i,t)$
for $(q = 1; q \leq n; q++)$ **do** ▷ n⟵Total appliances in $\mathbb{k}$
    while $(E_{def}^{grid}(t) > 0)$ **do**
        Extract associated home (in h) from list $\mathbb{k}$
        Switch OFF appliance q
        Communicate this decision to the local controller of h
        Update $E_{def}^{grid}(t) = E_{def}^{grid}(t) - E_q$
        Update $E_c(i)$ using Eq. (5.14)
    end while
end for

5.4.3 Managing instantaneous load changes

As the loads are dynamic in nature and user can override the control of loads in a smart home at any time, thus it is essential for any demand response management scheme to manage instantaneous changes in the load requirements. To handle the same, algorithm 8 is employed by the proposed scheme which is described as given below.

Whenever a new device d (with energy E_d) is switched ON by the user in a smart home and value of $E_{def}^{grid} < 0$, it would be instantly accommodated by the utility. However, if $E_{def}^{grid} > 0$ and the user requires this device urgently, then to switch it ON, the load is managed locally in the smart home. The urgency of a device is computed from the priority signal (Sig) which is initialized as $'urgent'$ if a device is switched ON more than twice in a given time-slot. In such scenarios, the load in a smart home is curtailed on the basis of appliance priorities. For this purpose, API of all the appliances in running state for that smart home is extracted and sorted in a list $Temp$. The appliances from $Temp$ are then switched OFF one by one until their cumulative energy becomes equal to E_d. The information of such appliances

Algorithm 8 Managing instantaneous load changes in smart homes.

When a new device, d (with energy requirement, E_d) is switched ON
if $(E_{def}^{grid} < 0)$ **then**
  Switch ON d
  Remove d from list AP, if present
else if $(Sig ==$ '$urgent$'$)$ **then** ▷ Sig ←Priority signal
  for $(\forall j \in i$ where $S_j(i) == 1)$ **do** ▷ j ←Appliances, i ←smart home
    Get $API_j(i,t)$ and sort in ascending order in list $Temp$
  end for
  $\forall j \in Temp$
  Switch OFF appliances from $Temp$ until $\sum E_j \geq E_d$
  Store these appliances in list AP; $pn_j = 1$
  Switch ON the new device d
else
  Store d in list AP; $pn_d = pn_d + 1$
end if
When an existing device, o (with energy, E_o) is switched OFF
if $(Sig \neq$ '$urgent$'$)$ **then**
  Update $E_{def}^{grid} = E_{def}^{grid} - E_o$
else
  break; /*As the energy would be consumed locally*/
end if

is stored in list AP and their pn number is set to 1. If $Sig \neq urgent$ and $E_{def}^{grid} > 0$, the new appliance d is not switched ON and its information is stored in list AP.

When any existing device (with energy E_o) is switched OFF instantly by the user and priority signal is not $'urgent'$, the value of E_o is decremented from E_{def}^{grid}. Otherwise, this energy is consumed locally by the new appliance which has been switched ON. In this manner, the instantaneous changes in load requirement of a smart home are managed by the utility.

5.5 Increasing Consumer Participation

To increase consumer's participation in the proposed scheme and to compensate for the consumer discomfort, an incentive scheme is proposed in this section. The incentives are given to each smart home at the end of the day which can be utilized while paying their electricity bills.

The incentive provided to the consumer is primarily dependent upon three factors viz. savings of the utility for that day, actual load adjusted for the day and the user-specified CCF. The savings of utility ($Util^{sav}$) (in \$) is the price saved by the utility which would have otherwise incurred, if it has to fulfill the entire load demand of the users. The value of $Util^{sav}$ depends upon utility constraints and its calculation is not in the scope of this research work as the main focus is on demand response management. The value of actual load adjusted is the ratio of the modified load demand after using DADR scheme to the initial load demand without using any scheme. This value is represented by load adjustment factor (LAF) which specifies the percentage change in the load profile of a smart home by using proposed scheme. It is mathematically represented as,

$$LAF(i,t) = \left[1 - \frac{\sum\limits_{\forall t} H_u^{new}(i,t)}{\sum\limits_{\forall t} H_u(i,t)}\right] \times 100 \tag{5.16}$$

where, $H_u^{new}(i,t)$ is the revised load demand after using DADR scheme. The third factor in calculating the consumer incentive is the user-specified value of CCF. As this value is adjusted throughout the day by the user, average CCF on total electricity consumption is calculated using the equation given below.

$$CCF^{avg}(i) = \frac{\sum\limits_{\forall t} CCF(i,t) \times H_u(i,t)}{\sum\limits_{\forall t} H_u(i,t)} \tag{5.17}$$

where, $CCF^{avg}(i)$ is the average CCF on total electricity consumption of the day for i^{th} smart home. The value of this factor in four smart homes for an entire month is shown in Fig. 5.6.

The values of $CCF^{avg}(i)$ and $LAF(i,t)$ are used to calculate the adjustment value ($A^{val}(i)$) for i^{th} smart home which is further used to compute the consumer incentive. The consumer is given more incentive when his comfort gets violated (i.e., when $LAF > CCF$) to compensate for the discomfort. Thus, different equations are used to calculate the value of A^{val} as illustrated in the case study that follows.

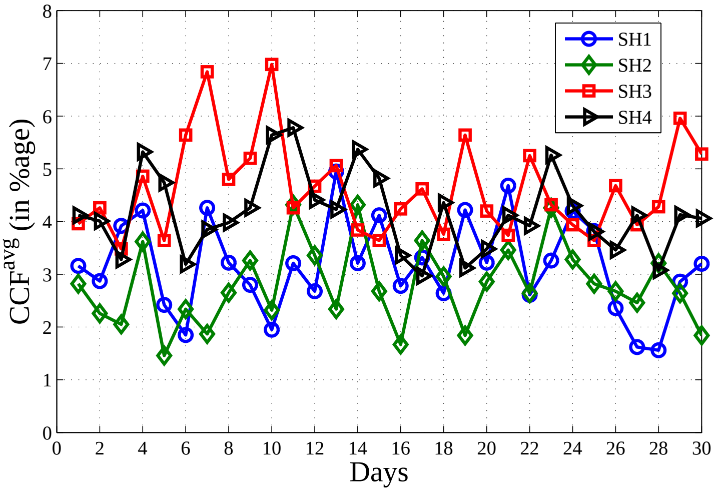


Figure 5.6: Average CCF of four smart homes for a month.

When $LAF(i) < CCF^{avg}(i)$, A^{val} is calculated as,

$$A^{val}(i) = \left(\frac{LAF(i).CCF^{avg}(i) + (CCF^{avg}(i) - LAF(i))}{2 \times CCF^{avg}(i)} \right) \tag{5.18}$$

When $LAF(i) \geq CCF^{avg}(i)$, A^{val} is calculated as,

$$A^{val}(i) = \left(\frac{LAF(i).CCF^{avg}(i) + (LAF(i) - CCF^{avg}(i))^2}{2 \times CCF^{avg}(i)} \right) \tag{5.19}$$

Case study: Let us consider the values of (LAF, CCF) in four different smart homes $\{SH1, SH2, SH3, SH4\}$ as (4%, 5%), (4%, 10%), (5%, 4%) and (7%, 4%) respectively. Now, even though the LAF for $SH1$ and $SH2$ is same, the A^{val} for $SH2$ should be more than $SH1$. This is because $SH2$ has given more liberty to the utility in load curtailment by specifying higher value of CCF. However, the A^{val} for $SH3$ and $SH4$ should be the higher than that of $SH1$ and $SH2$ as the load curtailed by the utility is more than their specified CCF. Using Eq. (5.18) for $SH1$ and $SH2$, and Eq. (5.19) for $SH3$ and $SH4$, the value of A^{val} for these SHs comes out to be $\{2.1, 2.3, 2.625, 4.625\}$ respectively. It can be noted that A^{val} for $SH4$ is more than $SH3$ as it has the more LAF as compared to $SH3$ for same value of CCF.

After computing A^{val} and receiving $Util^{sav}$ from the utility, the consumer incentive is calculated as,

$$CI(i) = A^{val} \times Util^{sav} \times \beta \tag{5.20}$$

where, $CI(i)$ is the incentive given to i^{th} smart home (in \$points) and β is the scaling constant. This incentive results in indirect savings for the users at a value decided by the utility. Considering that \$1 is paid for every 10000 accumulated incentive points, the total indirect saving for the consumer of i^{th} smart home can be calculated as,

$$Price_{sav}^{indirect}(i) = \frac{CI^{tot}(i)}{10000} \tag{5.21}$$

where, $Price_{sav}^{indirect}(i)$ is the indirect savings redeemed in lieu of incentives and $CI^{tot}(i)$ is the total accumulated $CI(i)$ in the month.

5.6 Performance Evaluation of DADR Scheme

The proposed scheme is tested in a simulated environment to validate its effectiveness in reducing peak load demand on the grid. The proposed scheme is evaluated by considering the time-slots of 15 minutes duration. The dataset of smart homes is taken from [129] which provides the hourly energy consumption values in a home for various purposes such as-lighting, heating, cooling, water heater, basic facilities and miscellaneous. These values are then tuned for 15 minute time intervals and distributed to corresponding appliances according to the usage pattern given by U.S. Energy Information Administration [145]. Fig. 5.7 shows the consumption pattern of different appliances in a smart home on a typical day of summer. The miscellaneous load curve shown in this figure includes the appliances like television, computer, washing machine, dishwasher and microwave. The other parameters used in this simulation study are given in Table 5.1. The cost of electricity $\wp$ is taken

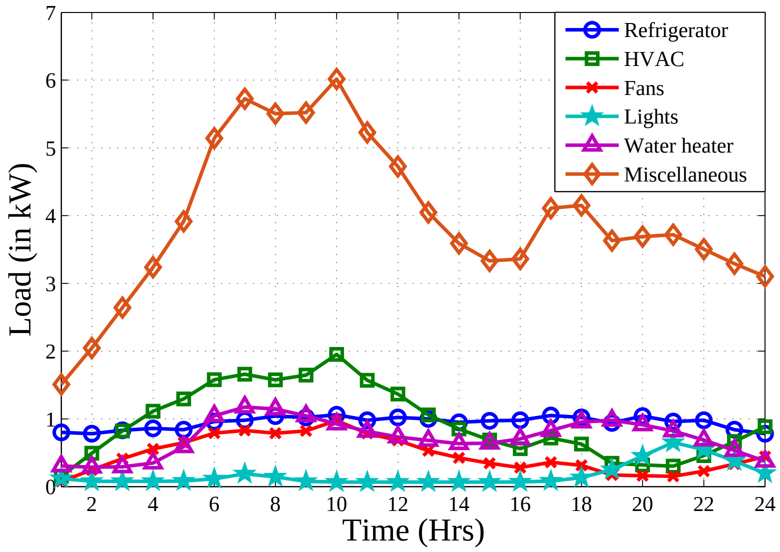


Figure 5.7: Load demand of different appliances.

from [146] while rest of the parameters are computed statistically. The value of CCF is randomly assigned to each household for simulation purposes.

Table 5.1: Various simulation parameters.

Parameters	**Values**
Δt	15 min.
τ	0.5
ϑ	50
γ	0.01
C	4
β	5
$\wp$	0.15 \$/kWh

5.6.1 Load reduction at grid

The main purpose of the proposed DADR scheme is to reduce the peak load. In order to evaluate the working of the proposed scheme, two case studies are presented. The first case study tests its performance in a realistic scenario; while the other is used to test it in a relatively harsh scenario.

Case study 1 - Load reduction in realistic scenario:

In this case study, the generation [147] and load [139] data on the grid are taken from PJM's real electricity market scenario for a typical day of summer. For simulation purpose, these data values are assumed to be in kWh and the generation data is further scaled down by 500 times; while the consumption of PJM's load zone of AEP IMP is scaled down by 10 times. This load zone is considered to comprise 50 smart homes from [129] which conform to the consumption pattern of the load zone. The final values of the generation and consumption data are depicted in Fig. 5.8.

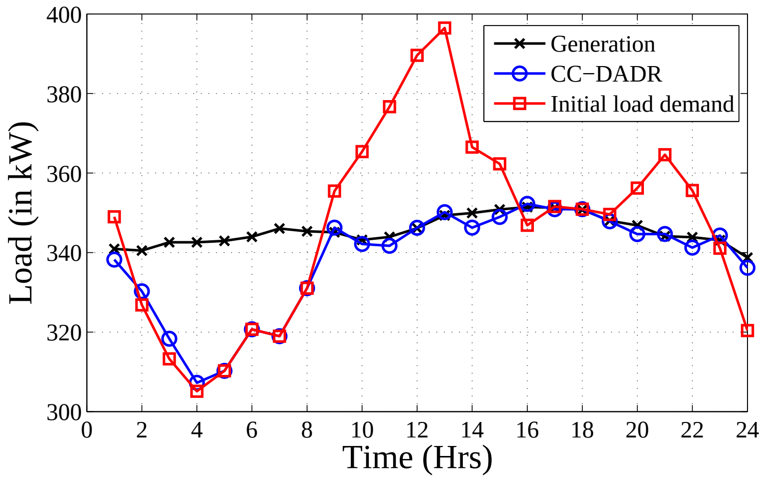


Figure 5.8: Demand curve for case study 1.

To reduce the peak in load demand, algorithm 6 is applied which is primarily based on calculating the value of API. For this purpose, homes exhibiting similar characteristics in terms of H_u are clustered together for each time-slot as discussed in Section 5.3.4. The number of clusters for 50 smart homes comes out to be two (This number may vary in accordance with the total number of smart homes considered in demand response management). On the basis of cluster formation, the value of API of all the appliances is computed using Eq. (5.9). An instance of API calculation for different appliances in three different smart homes in this scenario at a particular time-slot is shown in Fig. 5.9. Once the API values of the appliances are calculated, CC-DADR algorithm is applied to reduce the load demand at the grid. Fig. 5.8 depicts the load demand at the grid after utilizing this algorithm. It can be seen from this figure that CC-DADR algorithm effectively brings down the peak load

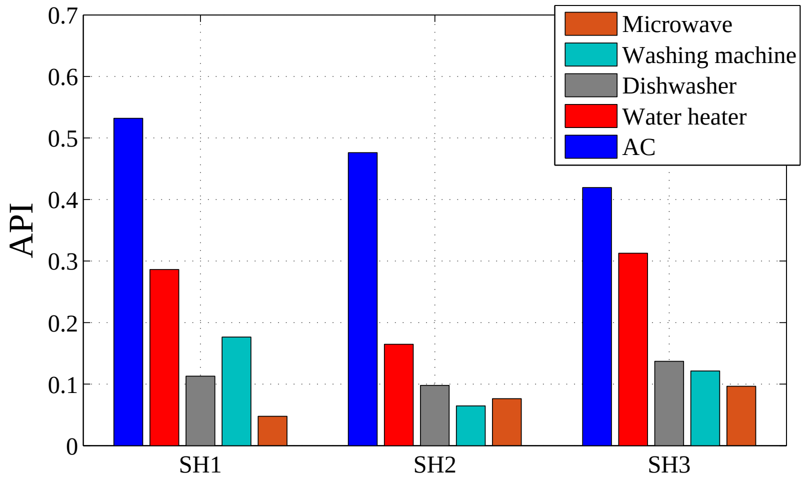


Figure 5.9: API of different appliances.

demand to the generation capacity of the grid. However, it may be the case in some scenarios that CC-DADR algorithm alone is not sufficient to reduce the peak load demand as discussed in the next case study.

Case study 2 - Load reduction in a harsh scenario:

This case study tests the efficacy of proposed scheme for large number of smart homes. In this case study, 200 smart homes from [129] are considered whose load requirements are assumed to be served during all the time-slots by a micro-grid of 1 MW power generation (i.e., $G(t) = 1$ MW, $\forall\ t$). This assumption is taken for the sake of simplicity as the primary focus of this scheme is from the viewpoint of managing the demand response in smart homes. Based on the demand of all the smart homes in an area, the initial load demand is computed as shown in Fig. 5.10. It can be seen in this figure that the load demand is often more than the power generation. This type of scenario is taken so as to test the proposed scheme in the environment where only the limited resources are available to suffice the load demand of the users. It can be seen in Fig. 5.10 that there are several peaks in load demand during 0800 to 1000 hours, 1300 to 1630 hours and 1900 to 2230 hours respectively. During such time-slots, the proposed scheme is applied as described in Section 5.4. Initially, the clustering of smart homes is performed to compute the value of API for every appliance of all the smart homes. For the number of smart

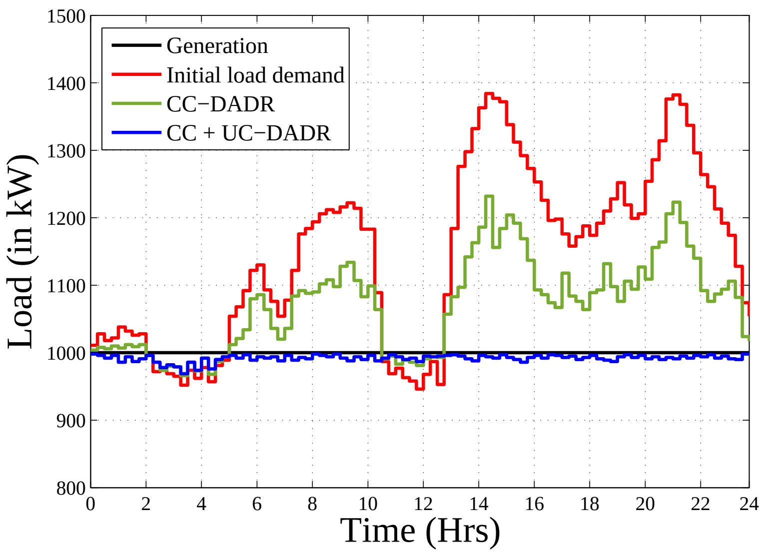


Figure 5.10: Demand curve for case study 2.

homes considered in this case study, the number of clusters are taken to be 4 (i.e., $C = 4$) as shown in Fig. 5.3. On the basis of cluster formation, the values of API and ACP of all the appliances are computed using Eqs. (5.9) and (5.11) respectively. Apart from these, the values of other parameters are also computed for managing the demand response as mentioned in Table 5.2.

For this purpose, the utility first employs algorithm 6 and if the peak in load is not reduced, then utility applies algorithm 7 to further reduce the peak in load demand. The load demand at the grid after applying CC-DADR algorithm alone and both the algorithms in tandem is shown in Fig. 5.10. It can be inferred from the figure that CC-DADR algorithm alone is not able to reduce the peak in load in this scenario. For example, the initial load demand at 1400 hours was 1363 kW, but the load demand after applying CC-DADR algorithm comes down to 1186 kW which is still more than the generation capacity. In such cases, UC-DADR algorithm is used by the utility in order to further flatten the load profile of the grid. As a result, the revised load demand after applying algorithm 7 at 1400 hours comes down to 996 kW as seen in Fig. 5.10. It can also be depicted from the figure that the proposed scheme effectively reduces the average load at the micro-grid during the peak hours. The value of this reduction comes out to be 14.84% using CC-DADR algorithm

Table 5.2: Values of different parameters for 3 smart homes.

	Appliance	ACP	AAF	RCI	AUF	App_k
SH1	A.C.	0.43	1	0.64	0.161	0.103
	Microwave	0.1	0		0.242	0.155
	Dishwasher	0.08	1		0.175	0.112
	Washing machine	0.142	1		0.102	0.065
	Water heater	0.377	1		0.057	0.036
SH2	A.C.	0.397	1	0.79	0.179	0.141
	Microwave	0.087	1		0.198	0.156
	Dishwasher	0.136	0		0.142	0.112
	Washing machine	0.098	1		0.18	0.142
	Water heater	0.188	1		0.056	0.044
SH3	A.C.	0.322	1	0.76	0.105	0.079
	Microwave	0.082	1		0.202	0.153
	Dishwasher	0.102	0		0.114	0.086
	Washing machine	0.097	0		0.189	0.144
	Water heater	0.348	1		0.1	0.076

alone, while the same is 27.38% if both algorithms are used in-tandem. However, the implication of using UC-DADR algorithm is that the utility have to pay more incentives in order to compensate for the violation in user's comfort. Nevertheless, as the main focus of this research is to reduce the peak load demand, thus both CC and UC-DADR algorithms are used in tandem and subsequent results are depicted after using both the algorithms in this scenario.

5.6.2 Energy savings in smart homes

The peak load reduction at grid reduces the overall energy consumption in smart homes and as a result, user saves a direct price which is given by,

$$Price_{sav}^{direct}(i) = \{E_c(i) \times \wp \mid at\ t = 24\} \tag{5.22}$$

where, $Price_{sav}^{direct}(i)$ is the direct price saved for i^{th} smart home at the end of the day and $\wp$ is the cost of electricity (in \$/kWh). This saving, in turn, reduces the total electricity cost of the user for the day as given in Table 5.3.

Table 5.3: Electricity cost using DADR scheme (in \$).

	Without DADR scheme	With CC-DADR algorithm alone	With CC and UC-DADR algorithms
SH1	3.53	3.43	3.34
SH2	4.45	4.33	4.27
SH3	2.7	2.6	2.58
SH4	6.45	5.99	5.82
SH5	3.15	2.98	2.98

It can be inferred from the table that cost of electricity after using CC-DADR algorithm alone is more as compared to the case when both CC-DADR and UC-DADR algorithms are used. It is because the load in a smart home cannot be curtailed beyond the specified value of CCF in CC-DADR algorithm. For instance, the average CCF for $SH1$ for a day is 3.16%; however, the load curtailed from $SH1$ using CC-DADR algorithm is only 2.8%. In some cases ($SH5$ in this case), the electricity cost after using CC-DADR algorithm is same as using both the algorithms. Such a case arises when no appliances are curtailed from a smart home while implementing UC-DADR algorithm.

5.6.3 Consumer incentives

The incentives provided to the consumers at the end of the day are primarily dependent upon the values of CCF^{avg} and LAF. For a single smart home, these values (obtained after applying proposed scheme) are shown in Fig. 5.11. Based upon these values, A^{val} is computed by using Eq. (5.18) (when $LAF < CCF^{avg}$) and Eq. (5.19) (when $LAF \geq CCF^{avg}$). Once the A^{val} is computed, the incentive points given to consumer are calculated using Eq. (5.20). Fig. 5.12 depicts the values of these incentive points given to the users of three smart homes for an entire month. It can also be inferred from Fig. 5.12 that more incentive is given to the consumers at the instances where the value of LAF is more than CCF^{avg}. For calculating these incentives, it is assumed that utility uses both the algorithms and the value of $Util^{sav}$ is taken to be \$100 (this value may vary according to utility constraints

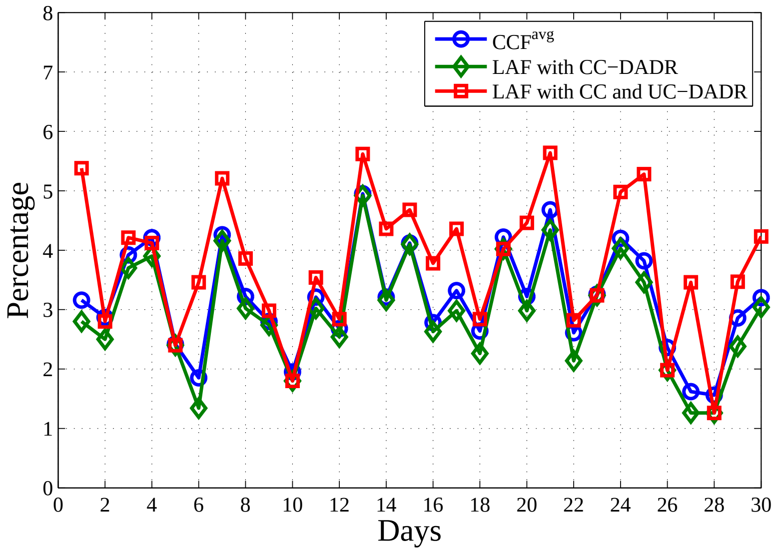


Figure 5.11: LAF after using DADR scheme.

in real scenarios).

5.6.4 Evaluation of proposed scheme for consumer satisfiability

The consumer is considered to be satisfied if he is able to save the cost on electricity after participating in DADR scheme without compromising his comfort. Thus, the two metrics, i.e., consumer savings and user comfort, are chosen to evaluate the consumer satisfiability in the proposed scheme.

1. Consumer savings:

The consumer earns two types of savings in lieu of his participation in the proposed scheme. First is the savings on direct price due to reduced electricity consumption in smart homes using the proposed scheme. These savings are computed at the end of each day using Eq. (5.22) as mentioned in Section 5.6.2. Second is the indirect price saved which is remitted to the consumer in the form of incentives. This saving is calculated at the end of the month using Eq. (5.21) and can be used while paying the electricity bills. In the proposed scheme, the indirect price saved by the five

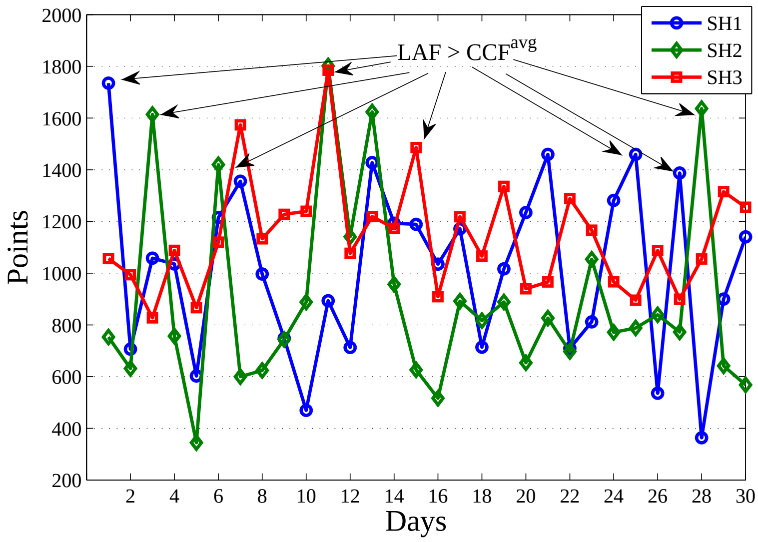


Figure 5.12: Incentive points given to consumers.

smart homes $\{SH1, SH2, SH3, SH4, SH5\}$ is calculated using Eq. (5.21) and it comes out to be {3.05, 2.68, 3.42, 3.92, 2.84} (in \$) respectively.

2. User comfort – LAF vs CCF:

User comfort is ensured by the proposed DADR scheme in two ways. Firstly, the load is curtailed by observing the usage trend of appliances in smart homes with similar load conditions. Secondly, the proposed scheme tries to keep the load curtailment in a smart home below the CCF value (which is specified directly by the user according to his comfort). This is evident from Fig. 5.13 where the value LAF is always below the value of CCF using CC-DADR algorithm. However, if the utility follows both the algorithms, there may arise few cases where the value of $LAF > CCF$. In such cases, an increased incentive is provided to the consumer (as given by Eq. (5.19)) to compensate for comfort violation. Thus, it can be said that the proposed scheme accounts for user comfort in the best possible way while respecting the utility conditions.

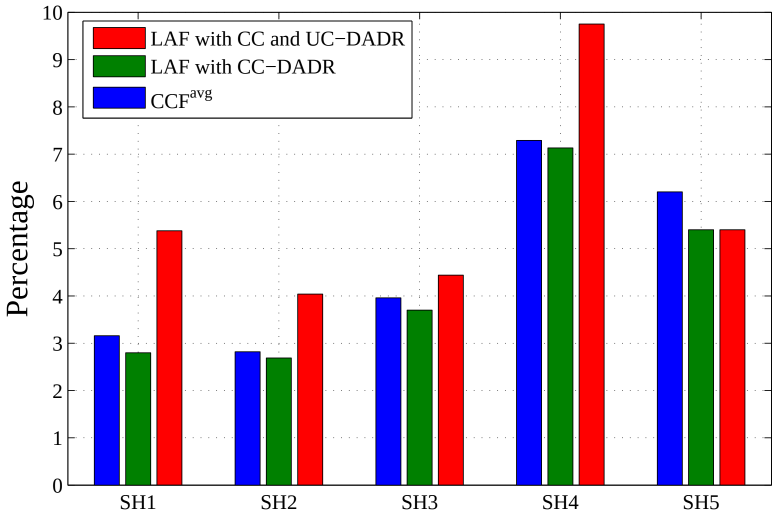


Figure 5.13: LAF vs CCF^{avg}.

5.6.5 Comparison with the existing schemes

The results of proposed scheme are compared with various existing schemes for the overall load reduction at the grid and the cost savings in smart homes in the similar test settings. The results of which are summarized in Table 5.4. It can be inferred from the table that the cost savings for proposed scheme are equivalent to that of [96]; while the proposed scheme has more value of load reduction at grid as compared to other approaches. Furthermore, the proposed scheme is tested on more number of smart homes and considers larger set of devices in smart homes when compared with its counterparts. Therefore, it can be said that the proposed scheme is more reliable for large scale implementation.

Table 5.4: Comparison with existing schemes.

Scheme	Load reduction at grid	Cost saving in a smart home	Number of smart homes considered	Devices considered in a smart home
[96]	20%	7.4%	5	7
[97]	19%	-	30	5
Proposed	27.38%	6.2%	200	16

5.7 Summary

The issue of peak load reduction at smart grid has been addressed in this chapter by designing a novel DADR scheme. Based on the data gathered from smart homes, various factors have been proposed which helps in taking demand response decisions. The proposed scheme can be viewed as a two-level control scheme in which the utility initially curtails the load in homes such that the user comfort is not violated (using CC-DADR algorithm alone). If the load demand is still more than the utility's desired level, then, second level of control comes into play which further reduces the peak load (using UC-DADR algorithm). However, in doing so, the user's comfort may get violated, in lieu of which the increased incentive has been provided to the consumer. In addition to it, the proposed scheme is also able to handle the instantaneous load changes in energy consumption pattern of smart homes. The proposed scheme has dual benefits; firstly, the proposed scheme effectively manages peak load demand in smart grid and secondly, the consumer is also benefited in terms of reduced electricity bills because of direct and indirect price savings.

Chapter 6

Data Processing and Management

In this chapter, a tensor-based data management technique is proposed to reduce the dimensionality of data gathered from the Internet-of-Energy (IoE) environment in a smart city [1] The core data is extracted from the gathered data by using tensor operations such as-matricization, vectorization and tensorization with the help of higher-order singular value decomposition. After reducing dimensionality of data, it can be used for providing many services in smart cities and its application to provide demand response services is discussed in this chapter. For this purpose, SVM-based classifier is used to classify the end-users (residential and commercial) into normal, overloaded and underloaded categories from the core data. Once such users are identified to take part in demand response mechanism, utilities can devise particular solutions to handle their demand response in order to alter their load requirements so that the overall load in the smart city is optimally managed. The results obtained on Open Energy Information and PJM dataset clearly indicate the supremacy of the proposed tensor-based scheme over the traditional scheme to classify the end-users.

[1] The contents of this chapter are partly published in: A. Jindal, N. Kumar, and M. Singh, "A unified framework for big data acquisition, storage and analytics for demand response management in smart cities," *Future Generation Computer Systems*, 2018, DOI: 10.1016/j.future.2018.02.039.

6.1 Contributions

The major contribution of this chapter are as follows.

1. A data acquisition framework to gather and process the data with respect to energy consumption in an IoE network in a smart city.

2. A tensor-based data representation scheme to manage and store the data collected in IoE paradigm.

3. A SVM-based classification scheme to identify the end-users in a smart city whose demand response can be managed in order to maintain the city's overall load profile.

6.2 Proposed Scheme

To manage the demand response of loads in a smart city, data about the loads connected to the IoE network needs to be stored in an effective way so that it can be processed easily. The framework for gathering such data in the IoE network has been shown in Fig. 6.1. In this figure, it can be seen that the complete data gathering and processing phase has been divided into three layers viz. data acquisition, data transmission and data processing. The first layer is the data acquisition layer where the data from different entities is gathered using the help of sensors installed at these entities. These entities include generation, transmission and distribution units, and end users such as smart homes, commercial loads including charging stations and industries. Different types of sensors such as smart meters, phasor measurement units (PMUs), advanced metering infrastructure (AMI), intelligent electronic devices (IEDs), etc. are deployed on these entities to gather different types of data related to energy.

The second layer is the data transmission layer that forwards the data sensed by the sensors to the data processing layer with the help of access points (APs).

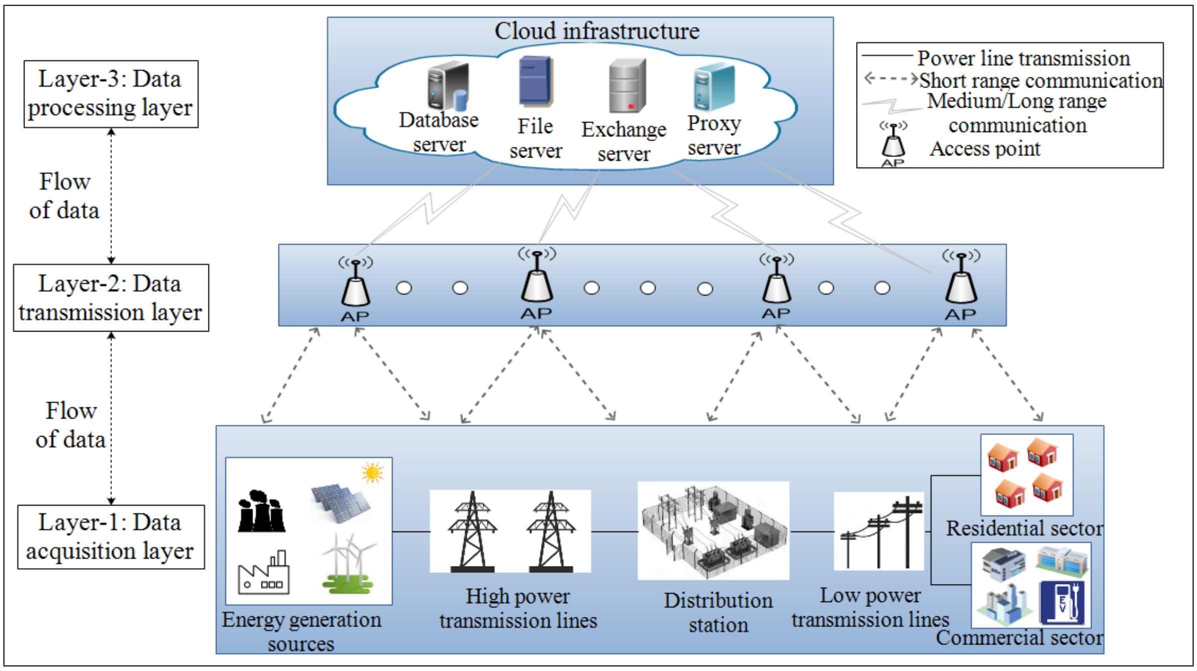


Figure 6.1: Data gathering and processing architecture in smart city.

These APs can be in the present in the form of roadside units or wireless routers deployed at various locations in the city to provide communication connectivity in IoE network [148]. For this purpose, small, medium and long range communication technologies are used as given in Table 6.1.

Table 6.1: Communication protocols used.

Communication	***Technologies***	***Protocols used***	***Frequency bands***	***Data rate***
Short range	DSA	IEEE 802.11af	476-494 MHz	1 Mbps
	DRSC/WAVE	IEEE 802.11p	5.850-5.925 GHz	3-27 Mbps
Medium/Long range	Wi-Fi	IEEE 802.11 a/b/g	2.4 - 5 GHz	1-54 Mbps
	WiMAX	IEEE 802.16	1.25 - 20 MHz	30 Mbps - 1Gbps
	LTE/LTE-A	-	20 - 100 MHz	300 Mbps - 3 Gbps

DSA - Dynamic Spectrum Access
DSRC - Dedicated short-range communication
WAVE - Wireless Access in Vehicular Environment
WiMAX - Worldwide Interoperability for Microwave Access
LTE - Long Term Evolution
LTE-A - Long Term Evolution Advanced

The third layer is the data processing layer which is used to store and process all the data gathered from the first layer. For storing and processing this data, cloud is one of the most viable options. The benefit of using cloud server for storing the data is that it has abundant resources which can be accessed at anytime from anywhere. Thus, it can continuously process the sensed data and the utility can also access the data pertaining to various entities at any time. Moreover, the cloud platform can easily be scaled if the data entries increase rapidly. Once the data is gathered at the utility server, it is stored in tensor form so as to reduce its dimensionality and then it is processed to apply demand response mechanism. For managing the demand response, the SVM classifier is used on this data to identify the users whose demand needs to be managed in order to maintain the overall load in a smart city. The phases related to tensor-based data dimensionality reduction and demand response mechanism are discussed in detail in the next sections.

6.3 Tensor-based Data Processing

The tensor-based data processing is required to efficiently store and process the data gathered from IoE network in the smart city. In order to store and represent the data in a manageable form, tensor-based approach to reduce the dimensionality of data is applied. The overall architecture of tensor-based processing of this data is shown in Fig. 6.2. Once the data is acquired from the data acquisition layer (as given in Fig. 6.1), this data is represented in the tensor format where data decomposition takes place to reduce its dimensionality. The initial data tensor is usually very large and cannot be stored in memory as it is, thus it becomes necessary to break it into small tensor blocks and store those blocks separately.

6.3.1 Basics of tensor

Tensor is a generalized multi-dimensional form of a standard matrix (a matrix can be considered to as a subset of tensor). A matrix has two modes, rows and columns (which makes it a 2^{nd} order tensor) whereas a N mode matrix is called N^{th} order

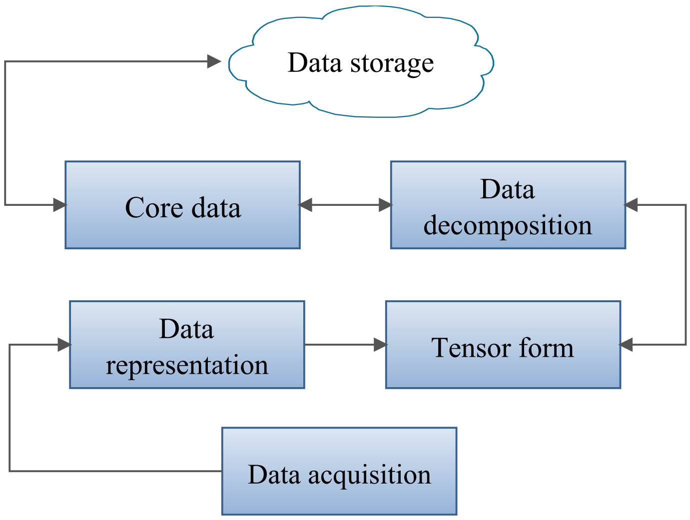


Figure 6.2: Tensor-based data processing.

tensor. A diagrammatic description of a 3^{rd} order tensor is represented in Fig. 6.3. In this figure, the notations are used in such a way that boldfaces and underlined capital alphabets represent the tensor (for example, $\underline{X} \in R^{I_1 \times I_2 \times \cdots \times I_n}$). Matrices ($2^{nd}$ order tensors) are depicted by boldfaced and capital alphabets ($\mathbf{X} \in R^{I \times J}$) and vectors ($1^{st}$ order tensors) by lower boldfaces alphabets ($\mathbf{x} \in R^{I}$).

It is to be noted that the number of elements grows exponentially with the order of tensor, thereby increasing the overall dimensionality of the tensor which would require a large number of computational and memory resources. Thus, it is important to reduce these dimensions in such a way that the overall resources required to handle a tensor become manageable. This can be achieved by decomposing the main tensor into small-scale core tensors and matrices which form the main building blocks of tensor networks. The advantage of using core tensors is that they are very easy to handle and visualize, and also help in compression of the large, incomplete and raw data. To understand more about tensors and its operations, following definitions are given [3, 116].

Definition 1 – Kronecker Product ($\otimes$): The Kronecker product of matrix $\mathbf{P} \in$

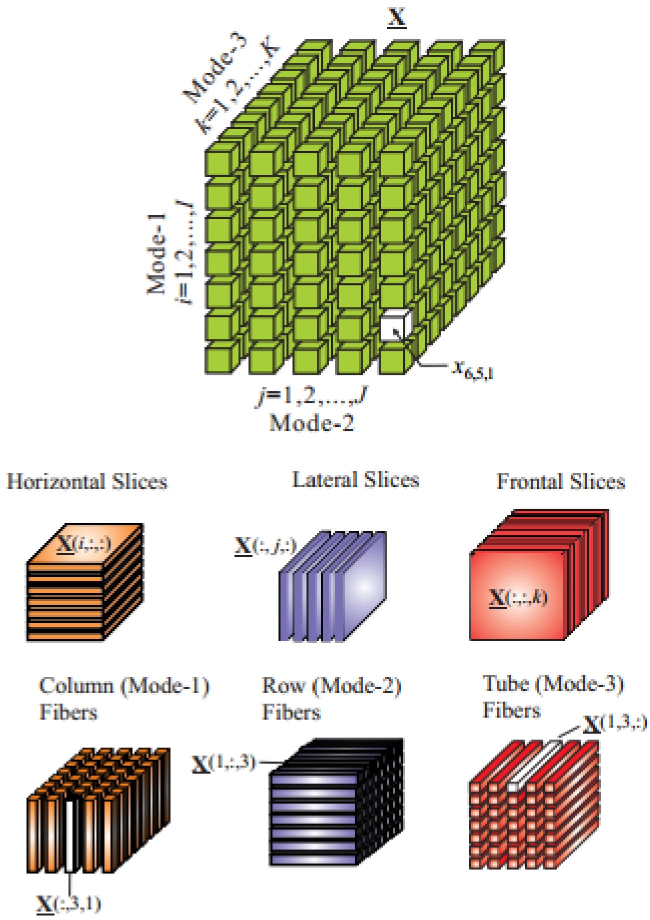


Figure 6.3: A 3^{rd} order tensor representation [3].

$R^{i\times j}$ and $\mathbf{Q} \in R^{k\times l}$ is given by:

$$\mathbf{P} \otimes \mathbf{Q} = \begin{bmatrix} p_{11}\mathbf{Q} & p_{12}\mathbf{Q} & \dots & p_{1j}\mathbf{Q} \\ p_{21}\mathbf{Q} & p_{22}\mathbf{Q} & \dots & p_{2j}\mathbf{Q} \\ \vdots & \vdots & \ddots & \vdots \\ p_{i1}\mathbf{Q} & p_{i2}\mathbf{Q} & \dots & p_{ij}\mathbf{Q} \end{bmatrix} \tag{6.1}$$

The matrices $\mathbf{M} \in R^{m\times n}$, $\mathbf{N} \in R^{p\times q}$, $\mathbf{O} \in R^{n\times o}$ and $\mathbf{P} \in R^{q\times r}$ in Kronecker product exhibits following properties:

$$(\mathbf{M} \otimes \mathbf{N})^T = \mathbf{N}^T \otimes \mathbf{M}^T \tag{6.2}$$

$$(\mathbf{M} \otimes \mathbf{N})(\mathbf{O} \otimes \mathbf{P}) = (\mathbf{MO} \otimes \mathbf{NP}) \tag{6.3}$$

$$(\mathbf{M} \otimes \mathbf{N}) \otimes \mathbf{O} = \mathbf{M} \otimes (\mathbf{N} \otimes \mathbf{O}) \tag{6.4}$$

Another popular variant format of standard Kronecker product is Left Kronecker product ($\otimes_L$) which is given by:

$$\mathbf{P} \otimes_L \mathbf{Q} = \begin{bmatrix} \mathbf{P}q_{11} & \mathbf{P}q_{12} & \dots & \mathbf{P}q_{1l} \\ \mathbf{P}q_{21} & \mathbf{P}q_{22} & \dots & \mathbf{P}q_{2l} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}q_{k1} & \mathbf{P}q_{k2} & \dots & \mathbf{P}q_{kl} \end{bmatrix} \tag{6.5}$$

Considering tensors rather than matrices, the Left Kronecker product of tensors $\underline{\mathbf{P}} \in R^{I_1 \times \cdots \times I_n}$ and $\underline{\mathbf{Q}} \in R^{J_1 \times \cdots \times J_n}$ forms a tensor $\underline{\mathbf{R}} = \underline{\mathbf{P}} \otimes_L \underline{\mathbf{Q}} \in R^{I_1 J_1 \times \cdots \times I_n J_n}$. The tensor $\underline{\mathbf{R}}$ yields same order as of other two tensors, but is larger in size.

Definition 2 – Khatri-Rao Product ($\odot$): The Khatri-Rao product of matrix $\mathbf{P} \in R^{i \times j}$ and $\mathbf{Q} \in R^{k \times j}$ is given by:

$$\mathbf{P} \odot \mathbf{Q} = [p_1 \otimes q_1 \; p_2 \otimes q_2 \; \cdots p_j \otimes q_j] \tag{6.6}$$

where, the output matrix has same number of columns as of original matrices and can be viewed as column-wise Kronecker product of the two matrices.

The matrices $\mathbf{M} \in R^{m \times n}$, $\mathbf{N} \in R^{p \times n}$ and $\mathbf{O} \in R^{q \times n}$ in Khatri-Rao product exhibits following properties:

$$(\mathbf{M} \odot \mathbf{N}) \odot \mathbf{O} = \mathbf{M} \odot (\mathbf{N} \odot \mathbf{O}) = \mathbf{M} \odot \mathbf{N} \odot \mathbf{O} \tag{6.7}$$

$$(\mathbf{M} \odot \mathbf{N})^T (\mathbf{M} \odot \mathbf{N}) = (\mathbf{M}^T \mathbf{M}) \circledast (\mathbf{N}^T \mathbf{N}) \tag{6.8}$$

Definition 3 – Hadamard Product ($\circledast$): The Hadamard product performs the element-wise scaler multiplication of two matrices of same size. For matrix $\mathbf{P} \in R^{i \times j}$

and $\mathbf{Q} \in R^{i \times j}$, the Hadamard product is given by:

$$\mathbf{P} \circledast \mathbf{Q} = \begin{bmatrix} p_{11}q_{11} & p_{12}q_{12} & \cdots & p_{1j}q_{1j} \\ p_{21}q_{21} & p_{22}q_{22} & \cdots & p_{2j}q_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ p_{i1}q_{i1} & p_{i2}q_{i2} & \cdots & p_{ij}q_{ij} \end{bmatrix} \tag{6.9}$$

Definition 4 – Outer Product ($\circ$): The outer product is another main operator in tensor analysis. For tensors $\underline{\mathbf{P}} \in R^{I_1 \times \cdots \times I_n}$ and $\underline{\mathbf{Q}} \in R^{J_1 \times \cdots \times J_m}$ forms a tensor $\underline{\mathbf{R}} = \underline{\mathbf{P}} \circ \underline{\mathbf{Q}} \in R^{I_1 \times \cdots \times I_n \times J_1 \times \cdots \times J_m}$.

Definition 5 – Mode-n Product ($\times_n$): The mode-n product (also known as tensor-times-matrix (TTM) product) of a tensor $\underline{\mathbf{A}} \in R^{I_1 \times \cdots \times I_n}$ with a matrix $\mathbf{B} \in R^{J \times I_n}$ is given by:

$$\underline{\mathbf{C}} = \underline{\mathbf{A}} \times_n B \in R^{I_1 \times \cdots \times I_{n-1} \times J \times I_{n+1} \times \cdots \times I_n} \tag{6.10}$$

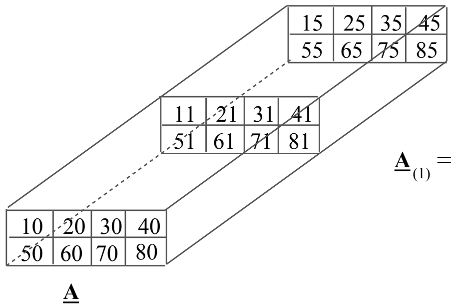
The mode-n product is the basic operation for reducing the dimensionality; it brings down the dimensionality of tensor from I_n to J ($J < I_n$).

Definition 6 – Matricization: The matricization (also known as matrix unfolding or flattening) re-arranges the elements of a tensor into a matrix with the help of two matricization operations.

mode-n matricization: The mode-n matricization of N^{th} order tensor $\underline{\mathbf{A}} \in R^{I_1 \times \cdots \times I_N}$ for index $n \in \{1, \cdots, N\}$ is given by:

$$\underline{\mathbf{A}}_{(n)} \in R^{I_n \times I_1 I_2 \ldots I_{n-1} I_{n+1} \ldots I_n} \tag{6.11}$$

where, number of rows of matrix are I_n and number of columns are $I_1 I_2 \ldots I_{n-1}$ $I_{n+1} \ldots I_n$. An example of the matricization of a 3^{rd} order tensor $\underline{\mathbf{A}} \in R^{2 \times 4 \times 3}$ is shown in Fig. 6.4.



$$\underline{\mathbf{A}}_{(1)} = \begin{bmatrix} 10 & 11 & 15 & 20 & 21 & 25 & 30 & 31 & 35 & 40 & 41 & 45 \\ 50 & 51 & 55 & 60 & 61 & 65 & 70 & 71 & 75 & 80 & 81 & 85 \end{bmatrix}$$

$$\underline{\mathbf{A}}_{(2)} = \begin{bmatrix} 10 & 50 & 11 & 51 & 15 & 55 \\ 20 & 60 & 21 & 61 & 25 & 65 \\ 30 & 70 & 31 & 71 & 35 & 75 \\ 40 & 80 & 41 & 81 & 45 & 85 \end{bmatrix} \quad \underline{\mathbf{A}}_{(3)} = \begin{bmatrix} 10 & 20 & 30 & 40 & 50 & 60 & 70 & 80 \\ 11 & 21 & 31 & 41 & 51 & 61 & 71 & 81 \\ 15 & 25 & 35 & 45 & 55 & 65 & 75 & 85 \end{bmatrix}$$

Figure 6.4: Matricization of a 3^{rd} order tensor.

mode-$\{n\}$ canonical matricization: For a N^{th} order tensor $\underline{\mathbf{A}} \in R^{I_1 \times \cdots \times I_N}$ with index $n \in \{1, \cdots, N\}$, the mode-$\{1, 2, \ldots n\}$ matricization is given by:

$$\underline{\mathbf{A}}_{<n>} \in R^{I_1 I_2 \ldots I_n \times I_{n+1} \ldots I_n} \tag{6.12}$$

where, number of rows of matrix are given by $I_1 I_2 \ldots I_n$ and number of columns are $I_{n+1} \ldots I_n$.

Definition 7 – Inner Product ($\langle\rangle$): The inner product of two tensors $\underline{\mathbf{A}}, \underline{\mathbf{B}} \in R^{I_1 \times \ldots I_n}$ can be reduced to inner product of two smaller tensor which is given by:

$$\langle \underline{\mathbf{A}}, \underline{\mathbf{B}} \rangle = vec(\underline{\mathbf{A}})^T vec(\underline{\mathbf{B}}) \tag{6.13}$$

where, $vec(\underline{\mathbf{A}})$ is vectorized form of $\underline{\mathbf{A}}$ which is represented as:

$$vec(\underline{\mathbf{A}}) = \left(\underset{n \in N}{\odot} \mathbf{X}^{(n)} \right) \lambda = \left(\underset{n \in N}{\otimes} \mathbf{X}^{(n)} \right) vec(\underline{\mathbf{G}}) \tag{6.14}$$

where, $\mathbf{X}^{(n)} = [\mathbf{x}_1^{(n)}, \mathbf{x}_2^{(n)}, \ldots, \mathbf{x}_R^{(n)}] \in R^{I_n \times R}$ are the factor matrices, $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_R]^T$ are the non-zero entries of the diagonal core tensor, $\underline{\mathbf{G}} \in R^{\mathbb{R}_1 \times \cdots \times \mathbb{R}_n}$ is the core tensor having $R_n << I_n$.

Definition 8 – Singular Value Decomposition (SVD): Another important feature in tensor operations is singular value decomposition (SVD) which forms the backbone of the low-rank matrix approximations. Let $\mathbf{X} \in R^{i \times j}$ be a matrix, then its SVD is given by:

$$\mathbf{X} = \mathbf{A}\mathbf{D}\mathbf{B}^T \tag{6.15}$$

where, matrices $\mathbf{A}$ and $\mathbf{B}$ denotes the column-wise orthogonal left and right singular vector space of X, respectively. The diagonal matrix $\mathbf{D} = diag\ (\sigma_1, \sigma_2, \ldots, \sigma_m, \ldots, \sigma_k)$ where $k = \min(i, j)$ contains the singular values of X. A much smaller and faster rank-r truncated SVD of $\mathbf{X}$ is calculated from the above equation which is given by:

$$\mathbf{X}_r = \mathbf{A}_r\mathbf{D}_r\mathbf{B}_r^T \tag{6.16}$$

where, $r < k$ and $\mathbf{A}_r = [a_1, a_2, \ldots, a_r]$, $\mathbf{B}_r = [b_1, b_2, \ldots, b_r]$ and $\mathbf{D}_r = diag(\sigma_1, \sigma_2, \ldots, \sigma_r)$.

Definition 9 – Frobenius Norm ($\|\ \|_F$): The frobenius norm of a tensor $\underline{\mathbf{A}} \in R^{I \times J \times K}$ can be computed in a simple manner if the factor matrices are orthogonal because its product becomes identity matrices.

$$\|\underline{\mathbf{A}}\|_F = \langle \underline{\mathbf{A}}, \underline{\mathbf{A}} \rangle = \sqrt{\sum_{I,J,K} \mathbf{A}(i,j,k)^2} \tag{6.17}$$

This norm has a similar role to singular values in standard SVD matrix.

The major tensor operations that are required to re-shape the tensor into matrices and vice-versa are given in Fig. 6.5. The matricization operations are performed a given in Eqs. (6.11) and (6.12), the vectorization operations are carried out using Eq. (6.14) and the tensorization operations are done with the help of $\otimes, \odot, \circledast, \circ, \times_n$.

The main purpose of using tensors is to reduce the dimension of the data so that

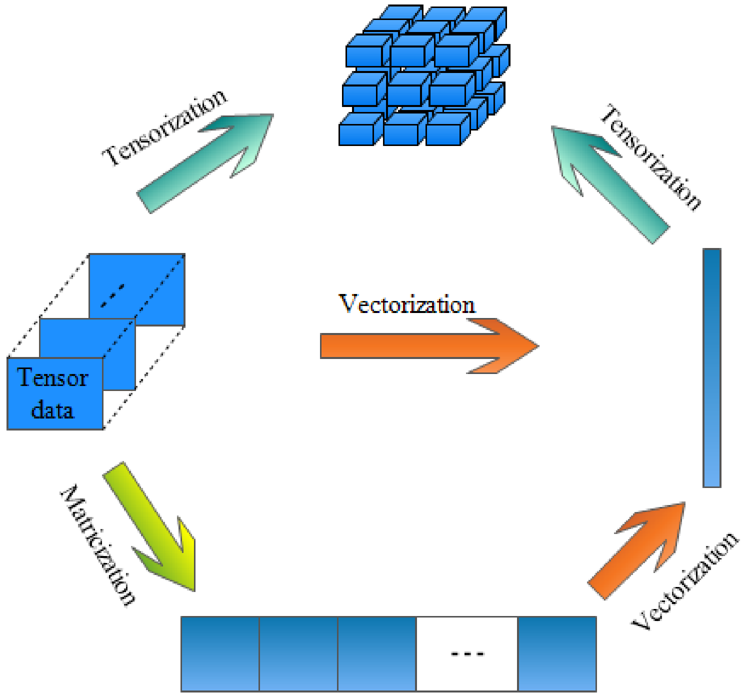


Figure 6.5: Tensor operations.

it can be represented and stored in an easy way. For this purpose, the data is compressed by decomposing it into various functions and wherein each function is stored separately. The original data is re-computed from these functions when required by performing tensor operations as discussed above. The concept of dimension reduction by decomposition can be understood as follows. A multivariate function can be approximated by a finite number of individual functions each representing one or more variables. For example, the function $f(x) = f(x_1, x_2, x_3, \ldots, x_n)$ can be represented separately as the following.

$$f(x_1, x_2, x_3, \ldots, x_n) \cong f^{(1)}(x_1)f^{(2)}(x_2)f^{(3)}(x_3)\ldots f^{(n)}(x_n) \tag{6.18}$$

In practical situations, the above decomposition is not possible with variables of higher order dimensions. Therefore, different tensor representation formats are used to separate these high dimensional functions. These formats are given by:

- **Canonical Polyadic (CP) format:** In this format, the tensor representation in discretized form is given by:

$$\underline{X} \cong \sum_{r=1}^{\mathbb{R}} \rho_r \mathbf{f}_r^{(1)} \circ \mathbf{f}_r^{(2)} \circ \cdots \circ \mathbf{f}_r^{(N)} \in R^{I_1 \times I_2 \times \cdots \times I_N} \tag{6.19}$$

where, ρ_r represents non-zero diagonal core tensor, $\mathbf{f}_r(n) \in R^{I_n}$, $\circ$ denotes the outer product, $\mathbb{R}$ is the tensor rank and N is the order of tensor. The CP format overcomes the problem of high dimension data representation, however, it may involve numerical problems for higher order tensors.

- **Tucker format:** In comparison to above, it gives more general factorizations of higher order tensors into small-sized core tensors. It is represented as follows.

$$\underline{\mathbf{X}} \cong \sum_{r_1=1}^{\mathbb{R}_1} \cdots \sum_{r_n=1}^{\mathbb{R}_n} \underline{\mathbf{G}}(f_{r_1}^{(1)} \circ f_{r_2}^{(2)} \circ \cdots \circ f_{r_n}^{(n)}) \tag{6.20}$$

where, $\underline{\mathbf{G}} \in R^{\mathbb{R}_1} \times \mathbb{R}_2 \times \cdots \times \mathbb{R}_n$ is the core tensor and $f_{r_n}^{(n)}$ represents the mode-n factor matrices. The Tucker format is more flexible as compared to CP format, but it is not practical for tensors of order more than 5 as the number of entries increase exponentially with increase in order size [3].

- **Tensor Train (TT) format:** In general, the TT format is given by:

$$f(x_1, x_2, x_3, \ldots, x_n) \cong \mathbf{F}^{(1)}(x_1)\mathbf{F}^{(2)}(x_2)\ldots\mathbf{F}^{(n)}(x_n) \tag{6.21}$$

where, $\mathbf{F}^{(n)}(x_n) \in R^{\mathbb{R}_{n-1} \times \mathbb{R}_n}$. The TT format is simple and most stable format to separate latent variables as it provides full control over low-ranked tensor network decompositions. Moreover, it shows very good control over the error of approximation which makes it more accurate as compared with other formats.

- **Hierarchical Tucker (HT) format:** It represents the tensors with the help of hierarchy of separations by making a binary tree which contains subset of modes at each branch. To understand it clearly, let us consider nested nonempty disjoint subsets a, b, and $t = a \cup b \subset \{1, 2, \ldots, n\}$, then for $1 \leq n_0 < n$, $a_0 = \{1, 2, \ldots, n_0\}$ and $b_0 = \{n_0 + 1, \ldots, n\}$, the tensor in HT format is represented as:

$$f(x_1, x_2, \ldots, x_n) \cong \sum_{r_{a_0}=1}^{\mathbb{R}_{a_0}} \sum_{r_{b_0}=1}^{\mathbb{R}_{b_0}} \underline{\mathbf{G}}\; f_{r_{a_0}}^{a_0}(\mathbf{x}_{a_0}) f_{r_{b_0}}^{b_0}(\mathbf{x}_{b_0}) \tag{6.22}$$

The dimension binary tree (T_N) using HT format is created such that it satisfies the following properties.

- All the nodes $t \in T_N$ are non-empty sub-sets of $\{1, \ldots, N\}$
- $t_{root} = \{1, \ldots, N\}$ is the root-node of T_N
- each non-leaf node has two branches $a, b \in T_N$ so that $t = a \cup b$.

6.3.2 Tensor decomposition

The main purpose of using tensors is that the data (with large dimensions) can be represented in terms of simple functions, i.e., core tensors and projection matrices (having small dimensions). The tensors can be represented using all the formats specified above according to the suitability of the problem. Considering the data gathered in IoE network, the Tucker format suits the best and data tensors are represented in Tucker form. In general, the core tensor $\underline{\mathbf{G}}$ can be formed from initial tensor $\underline{\mathbf{A}}$ with reduced dimensions as follows:

$$\underline{\mathbf{G}} = \underline{\mathbf{A}} \times_1 \mathbf{P}_1^T \times_2 \mathbf{P}_2^T \cdots \times_N \mathbf{P}_N^T \tag{6.23}$$

where, $\{\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_N\}$ are the projection matrices in N dimensions. An example of decomposition of a 3^{rd} order tensor $\underline{\mathbf{A}} \in R^{I \times J \times K}$ into core tensor $\underline{\mathbf{G}} \in R^{M \times N \times O}$ and projection matrices $\mathbf{B} \in R^{I \times M}$, $\mathbf{C} \in R^{J \times N}$ and $\mathbf{D} \in R^{K \times O}$ is shown in Fig. 6.6. Once the core tensor and projection matrices are computed and stored, the original tensor can be reconstructed when required with some approximation as follows.

$$\underline{\mathbf{A}} \cong \underline{\mathbf{G}} \times_1 \mathbf{P}_1 \times_2 \mathbf{P}_2 \cdots \times_N \mathbf{P}_N \tag{6.24}$$

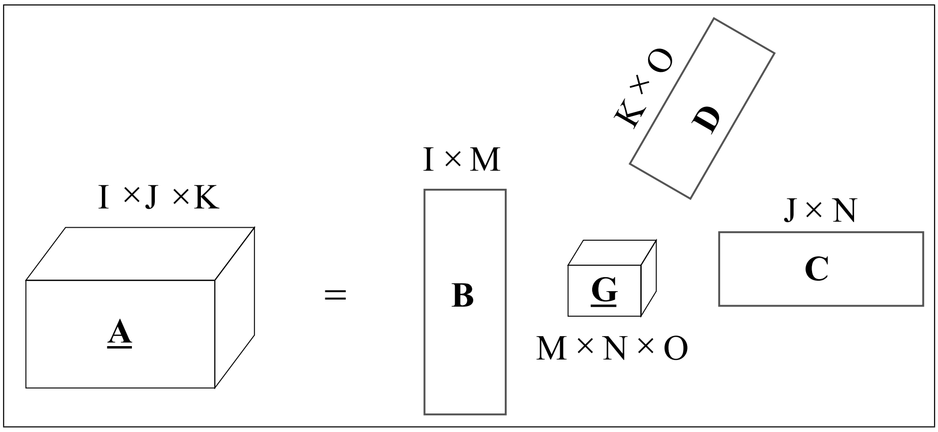


Figure 6.6: Tensor decomposition of a 3^{rd} order tensor.

All the tensor decomposition methods are based on the concept of SVD which is performed for higher order tensors. To perform this task, higher-order SVD-based decomposition algorithm is proposed to reduce the dimension of data as given in algorithm 9. Using this algorithm, the core data (i.e. core tensors and projection matrices) is extracted out of the original data which is stored on the utility server as shown in Fig. 6.2. The working of this algorithm is explained below.

The heterogeneous data collected at a particular time instant from the IoE network is represented in the form of low-rank tensors $\{\underline{\mathbf{T}}_1, \underline{\mathbf{T}}_2, \ldots, \underline{\mathbf{T}}_M\}$. A unified tensor $\underline{\mathbf{T}}$ is then created from the heterogeneous tensors by combining these tensors. After this step, the unified tensor is decomposed into SVD matrices in smaller dimensions and the mode-2, mode-1 cores are replaced with newly computed SVD matrices. Then, the $\delta-$truncated SVD mode-1 cores are calculated using Eqs. (6.16) and (6.17) such that the truncated matrices falls in the permissible tolerance limit ξ. The mode-2 and mode-1 tensor cores are again replaced using the truncated SVD matrices and the final unified tensor is computed with reduced cores. After this step, the core tensor $\underline{\mathbf{G}}$ and projection matrices $\mathbf{P}_{(n)}$ of $\hat{\underline{\mathbf{T}}}_{(n)}|_{n=1}^{N}$ are computed which are stored on the utility server. The advantage of using this algorithm for calculating

Algorithm 9 Higher-order SVD-based decomposition for dimension reduction.

Input: Heterogeneous data in terms of tensors $\{\underline{\mathbf{T}}_1, \underline{\mathbf{T}}_2, \ldots, \underline{\mathbf{T}}_M\}$, tolerance limit ξ

Output: N^{th} order unified tensor $\hat{\underline{\mathbf{T}}}$ with reduced dimensions such that $\|\underline{\mathbf{T}} - \hat{\underline{\mathbf{T}}}\|_F \leq \xi\|\underline{\mathbf{T}}\|_F$, Projection orthogonal matrices $\mathbf{P}_{(n)}|_{n=1}^{N}$ and core tensor $\underline{\mathbf{G}} \in R^{r_1 \times r_2 \times \cdots \times r_N}$

Create low-rank tensors $\underline{\mathbf{T}}_1, \underline{\mathbf{T}}_2, \ldots, \underline{\mathbf{T}}_M$ of the heterogeneous data
Compute a unified tensor $\underline{\mathbf{T}} = \langle \underline{\mathbf{T}}_1, \underline{\mathbf{T}}_2, \ldots, \underline{\mathbf{T}}_M \rangle \in R^{I_1, I_2, \ldots, I_N}$
Initialize $\hat{\underline{\mathbf{T}}} = \underline{\mathbf{T}}$ and $\delta = \frac{\xi}{\sqrt{N-1}}$
for $(n = 1 \ to \ N - 1)$ **do**
    Decompose $\mathbf{T}^{(n)}_{<2>} = \mathbf{U}_n \mathbf{V}; \mathbf{T}^{(n)}_{<2>} \in R^{\mathbb{R}_{n-1} I_n \times \mathbb{R}_n}$
    Replace cores $\mathbf{T}^{(n)}_{<2>} = \mathbf{U}_n$ and $\mathbf{T}^{(n+1)}_{<1>} \leftarrow \mathbf{V}\mathbf{T}^{(n+1)}_{<1>}$; $\mathbf{T}^{(n+1)}_{<1>} \in R^{\mathbb{R}_n \times I_{n+1} \mathbb{R}_n}$
end for
for $(n = N \ to \ 2)$ **do**
    Compute $\delta-$ truncated SVD $\mathbf{T}^{(n)}_{<1>} = \mathbf{B}\mathbf{D}\mathbf{C}^T$ using Eqs. (6.16) and (6.17)
    Replace cores $\hat{\mathbf{T}}^{(n-1)}_{<2>} \leftarrow \hat{\mathbf{T}}^{(n-1)}_{<2>} \hat{\mathbf{B}}\hat{\mathbf{D}}$ and $\hat{\mathbf{T}}^{(n)}_{<1>} = \hat{\mathbf{C}}^T$
end for
Combine N^{th} order tensor $\hat{\underline{\mathbf{T}}} = \langle \hat{\underline{\mathbf{T}}}_{(1)}, \hat{\underline{\mathbf{T}}}_{(2)}, \ldots, \hat{\underline{\mathbf{T}}}_{(N)} \rangle \in R^{I_1 \times I_2 \times \cdots \times I_N}$ with reduced cores $\hat{\underline{\mathbf{T}}}_{(n)} \in R^{\mathbb{R}_{n-1} \times I_n \times \mathbb{R}_n}$
for $(n = 1 \ to \ N)$ **do**
    $\underline{\mathbf{A}} \leftarrow \hat{\underline{\mathbf{T}}}_{(n)}$
    Do
    for $(j = 1 \ to \ N)$ **do**
        $\underline{\mathbf{X}} \leftarrow \underline{\mathbf{A}} \times_1 \mathbf{P}^T_{(1)} \cdots \times_{(i-1)} \mathbf{P}^T_{(i-1)} \times_{(i+1)} \mathbf{P}^T_{(i+1)} \cdots \times_n \mathbf{P}^T_{(j)}$
        Compute co-variance matrix of mode-j matricization using $\mathbf{C} = \mathbf{X}_{(j)} \mathbf{X}^T_{(j)} \in R^{\mathbb{R} \times \mathbb{R}}$
        Set $\mathbf{P}_{(j)}$ to be r_j left eigenvectors of $\mathbf{C}$.
    end for
    Compute core tensor using $\underline{\mathbf{G}} = \underline{\mathbf{A}} \times_1 \mathbf{P}^T_{(1)} \times_2 \mathbf{P}^T_{(2)} \cdots \times_{\mathbb{R}_n} P^T_{(\mathbb{R}_n)}$
    While $((\|\underline{\mathbf{A}}\|^2_F - \|\underline{\mathbf{G}}\|^2_F)$ converges)
    Store $\{\underline{\mathbf{G}}; \mathbf{P}_{(n)}|_{n=1}^{N}\}$ on the utility server
end for

the core tensors is that it finds the best approximation of core tensor which can be later used to recover the original tensor with minimum error. This is ensured using step 22 of algorithm 9 which defines that the algorithm works until the difference between the Frobenius form of the original tensor and core tensor converges to a minimum. Whenever the original data is required, it can be computed using Eq. (6.24) with the help of core tensor and projection matrices.

An Example:

To understand the concept of higher-order SVD for dimension reduction using algorithm 9, the following example (with respect to the 3^{rd} order tensor decomposition as represented in Fig. 6.6) is presented. Let us consider a random unified tensor $\underline{\mathbf{A}}$ of dimensions $5 \times 4 \times 3$ as given below.

$$\underline{\mathbf{A}}(\,:\,,\,:\,,\,\mathbf{1}) = \begin{bmatrix} 0.1250 & 0.2842 & 0.2465 & 0.2368 \\ 0.1636 & 0.3596 & 0.3086 & 0.3000 \\ 0.1714 & 0.3521 & 0.2956 & 0.2947 \\ 0.2021 & 0.4257 & 0.3605 & 0.3559 \\ 0.0407 & 0.0873 & 0.0744 & 0.0729 \end{bmatrix}$$

$$\underline{\mathbf{A}}(\,:\,,\,:\,,\,\mathbf{2}) = \begin{bmatrix} 0.5157 & 1.1723 & 1.0165 & 0.9765 \\ 0.6749 & 1.4832 & 1.2729 & 1.2374 \\ 0.7067 & 1.4520 & 1.2193 & 1.2153 \\ 0.8335 & 1.7560 & 1.4869 & 1.4679 \\ 0.1678 & 0.3601 & 0.3068 & 0.3008 \end{bmatrix}$$

$$\underline{\mathbf{A}}(\,:\,,\,:\,,\,\mathbf{3}) \;=\; \begin{bmatrix} 0.0233 & 0.0529 & 0.0458 & 0.0440 \\ 0.0304 & 0.0669 & 0.0574 & 0.0558 \\ 0.0319 & 0.0655 & 0.0550 & 0.0548 \\ 0.0376 & 0.0792 & 0.0670 & 0.0662 \\ 0.0076 & 0.0162 & 0.0138 & 0.0136 \end{bmatrix}$$

Let the core tensor $\underline{\mathbf{G}}$ of $\underline{\mathbf{A}}$ has the dimensions $3 \times 2 \times 1$ and projection matrices $\{\mathbf{B}, \mathbf{C}, \mathbf{D}\}$ are of lengths {(5,3), (4,2), (3,1)} respectively. Then, using the algorithm 9 for the tensor decomposition, the core tensor an projection matrices comes out to be:

$$\underline{\mathbf{G}} \;=\; \begin{bmatrix} 0.8147 & 0.9134 \\ 0.9058 & 0.6324 \\ 0.1270 & 0.0975 \end{bmatrix}$$

$$\mathbf{B} \;=\; \begin{bmatrix} 0.2785 & 0.9706 & 0.4218 \\ 0.5469 & 0.9572 & 0.9157 \\ 0.9575 & 0.4854 & 0.7922 \\ 0.9649 & 0.8003 & 0.9595 \\ 0.1576 & 0.1419 & 0.6557 \end{bmatrix}$$

$$\mathbf{C} \;=\; \begin{bmatrix} 0.0357 & 0.7577 \\ 0.8491 & 0.7431 \\ 0.9340 & 0.3922 \\ 0.6787 & 0.6555 \end{bmatrix}$$

$$\mathbf{D} = \begin{bmatrix} 0.1712 \\ 0.7060 \\ 0.0318 \end{bmatrix}$$

Complexity analysis: The overall complexity of algorithm 9 is dependent on SVD, matricization, unification and mode-n product operations for tensors. The SVD operations require the matrices to decompose into two matrices (of order say $k \times k$) and this step is repeated for N matrices which makes its complexity equal to $O(k^2N)$. The matricization and unification operations are simple transformation operations which take $O(1)$ time to execute. The mode-n product operation is a matrix-matrix product operation which takes $O(k^2N)$ time for two orthogonal matrices having k columns. So, the overall complexity of this algorithm comes out to be $O(k^2N) + O(1) + O(1) + O(k^2N)$ which is $O(k^2N)$.

6.4 Demand Response Management

Once the data is collected and stored on the utility server, it can be used for various purposes such as-outage detection, theft detection, demand response management, load forecasting, price prediction and many more applications. Many of these applications such as-load forecasting, demand response management, etc. make use of historical data to build their models. Thus, the storage of data becomes particularly important for these applications. However, the gathered data is enormous and heterogeneous in nature which cannot be processed as it is, thus it is necessary to remove unwanted and redundant data to reduce its dimension so as to analyze it quickly. For this purpose, the tensor-based dimensionality reduction of this data is performed as given in the previous section with an aim to provide demand response management in smart city environment. In order to manage the demand response, SVM classifier is employed on the tensor data to assign different labels to the end-users (residential and commercial users). The core tensors and projection matrices related to the required data are extracted from the server and the original data tensors are recovered from them. After this step, these data tensors are given as an input to the SVM classifier which labels the overall load in the city as normal, underloaded and overloaded. So, whenever there is a mismatch between demand and supply, the end-users' load in the city is labeled into three categories viz. normal, underloaded and overloaded on the basis of data collected from energy generation sources and end-users. Depending upon the output of SVM classifier, utility can further provide a different set of offers or solutions to increase (or decrease) the load profiles of underloaded (or overloaded) end-users so that the overall load profile of the city remains stable. One such solution is reported in Chapter 4 where we proposed a load balancing algorithm to manage the load of users classified as overloaded. Similar solutions can be incorporated with the proposed scheme to manage the load of users' classified as overloaded; while the underloaded users can sell their energy resources to other homes so as to earn profit and balance their load profiles. However for this to happen, the classifier needs to identify the end-users which are underloaded or overloaded based on their consumption patterns.

Algorithm 10 Training of SVM classifier

Input: Core data stored on the server
Output: SVM classification model

```
Get the dimensionally reduced core tensors and projection matrices of the related data from server storage
Construct the original data tensor T̂ with reduced dimensions using Eq. (6.24)
Extract the required features in matrices from T̂ using matricization operation
Convert the features matrices into feature vectors using Eq. (6.14)
Convert categorical features into numeric features
for (j = 1; j ≤ n; j++) do                                ▷ F_i ← Feature vector
    Normalize F_i in [-1,+1] using Eq. (3.9)
    Store normalized data in D_n(i)
end for
Select Gaussian RBF kernel function from Eq. (3.22)
Use grid search method to find initial modeling parameters (C, γ) in range (2^-10, 2^-8, ..., 2^8, 2^10)
Compute best C and γ using cross-validation and grid search methods
Select best (C, γ) to re-train the classifier on D_n
return SVMclassifier(F_i, C, γ)
```

The working of SVM classifier for assigning labels is explained as follows. As data recovered from the related entities do not have any inconsistent or missing values (due to tensor processing), no data pre-processing is required. However, the categorical data is converted into numeric format for its classification by the SVM. The overall working of SVM classifier for classification purposes has been elaborated in Chapter 3 and for the proposed scheme, it is summarized in algorithm 10.

6.5 Results for Tensor-based Data Processing

The results obtained after performing simulation are summarized in this section. As the focus of the scheme is to provide demand response services, the simulation scenario only considers the data related to load requirements of the end-users (residential & commercial users) and the generation of power from different sources as shown in Fig. 6.1. The data about generation of power is taken from PJM [147]. This data is scaled down for carrying out the simulation task. The residential sector is assumed to comprise of 1000 smart homes; while commercial sector consists of 200 buildings and industries like hospitals, schools, restaurants, supermarkets, hotels, warehouses, etc. The consumption data of these sectors is gathered from Open Energy Information dataset [129]. The data about season, weather and temperature is gathered from [130]. The sampling rate of data is assumed to be 1 second and feature vector for every end-user load comprises of 12 features. These features are namely overall load consumption, time-slot, season, weather, temperature, number of appliances, average load consumption at previous hour, average load consumption of previous day, average load consumption of previous week, weekend/weekday, number of occupants and local power generation. Considering all these features, the overall residential data comprises of $1000 \times 12 \times 86400$ entries while the commercial entries amass to $200 \times 12 \times 86400$. Storing and extracting value out of these entries is a problem which is solved using the proposed tensor-based decomposition.

6.5.1 Tensor-based processing of data

As the demand response decisions are often taken for a time-slot of 15 minutes by the utilities, the data tensor for 15-minute time intervals are formed for the end-users. Each tensor consists of $1000\times12\times900$ elements for residential users and $200\times12\times900$ elements for commercial users. Thus, the unified tensor consists of $1200 \times 12 \times 900$ entries in total. The data gathered from this huge IoE network in a smart city is processed using tensor-based operations and its dimensionality is reduced using algorithm 9 before storing it. The number of iteration for algorithm 9 are performed so as to evaluate the effects of reduced dimensionality. The reduction in overall dimensionality of the data tensor can be measured with the help of dimensionality reduction ratio [118] which is given by,

$$\vartheta = \frac{nze(\underline{\hat{\mathbf{G}}}) + \sum_{i=1}^{N} nze(\mathbf{U}_i)}{nze(\underline{\mathbf{T}})} \tag{6.25}$$

where, $nze(X)$ are the non-zero elements in X (which represents the initial tensor $\underline{\mathbf{T}}$, core tensor $\underline{\hat{\mathbf{G}}}$, projection matrices $\mathbf{U}_i$).

Once the data is stored in reduced form, it is required to be re-constructed to the initial form before being processed. This re-construction is carried out in a simple way with the help of Eq. (6.24). However, it is worth noting that the re-constructed tensor and the original tensor may vary due to the reduction in dimension. This variation can be measured with the help of re-construction error which is given by,

$$\psi = \frac{\|\underline{\mathbf{T}} - \underline{\hat{\mathbf{T}}}\|_F}{\|\underline{\mathbf{T}}\|_F} \tag{6.26}$$

where, ψ is the re-construction error degree in terms of Frobenius norm and $\underline{\hat{\mathbf{T}}}$ is the re-constructed tensor.

For the given tensor of dimension $1200 \times 12 \times 900$, the dimensionality reduction ratio with respect to number of iterations of algorithm 9 is given in Fig. 6.7. It

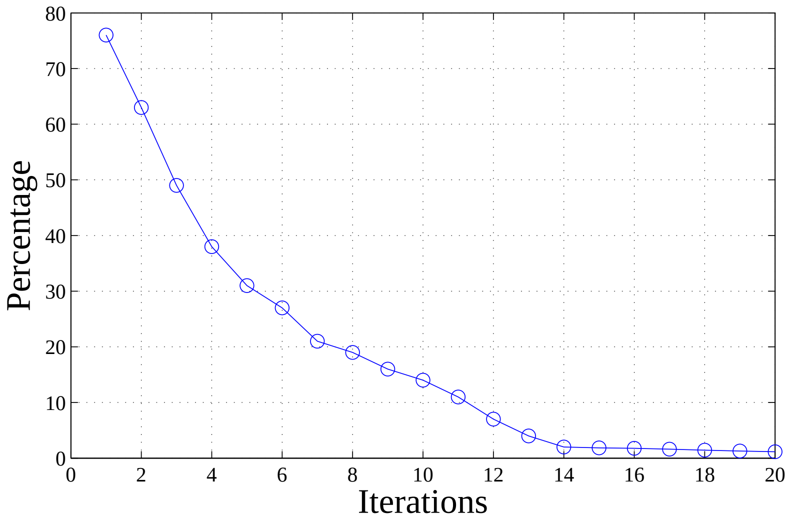


Figure 6.7: Dimension reduction ratio of the tensor.

can be seen in this figure that with each iteration, the dimension of the tensor is reduced with minimum decomposition in the first iteration (i.e. 76%) which further reduces to 14% in the tenth iteration and maximum decomposition happening in the fourteenth iteration after which it converges. However, this does imply that the algorithm should be executed for this number of iterations for reducing the dimension of the original tensor. It is because the re-construction error increases proportionally with respect to the dimension reduction ratio as evident from Fig. 6.8. This figures clearly indicates that the re-construction error is minimum in the

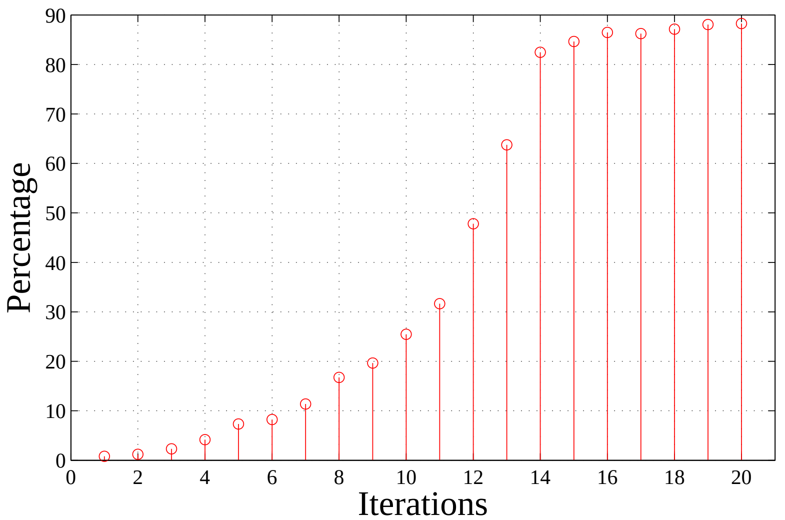


Figure 6.8: Re-construction error while forming original tensor.

first iteration with the value of 0.8% which increases to 25.4% in the tenth iteration and converges after the fourteenth iteration at 82.5%. From the first iteration, it can be inferred that 76% of the core data stored can give accuracy of 99.2% while constructing the original tensor for data processing and at the tenth iteration, 14% of core data can reconstruct the original tensor with 74.6% accuracy. This reconstruction accuracy further reduces to 17.5% with only 2% of the core data at the fourteenth iteration. Thus, it can be deduced that there should be a trade-off between dimensionality reduction and re-construction error so that minimum data is stored which gives maximum amount of recovery. For the given tensor, the trade-off between reduction ratio and re-construction error can be considered in the fourth iteration where 38% of core data gives the re-construction error of 4.16%.

6.5.2 SVM for demand response management

When the overall load demand is not equal to the power generated in the city, the demand response of the loads can be managed so as to maintain the balance between demand and supply. For this purpose, the SVM identifies the end-users (whose load demand can be altered) by assigning labels to these users. Based on these labels, the utility can opt certain strategies to lure such users that fall into the overload or underload utilization categories so as to optimally use the power generation resources in the smart city. The SVM classifier used for labeling the users is described in algorithm 10. Initially, the feature vectors are extracted from the core data which is stored on the server. After this step, the feature vectors are normalized in the range of [-1,+1]. Once the data values are normalized, various kernel functions are tried on the dataset to choose the best kernel function to be used for classification purposes. The comparison of accuracies for various types of kernel functions is given in Table 6.2. It is clear from this table that the Gaussian RBF outperforms all the other kernel functions. Therefore, the Gaussian RBF kernel function is used to train the classifier in the proposed scheme. In this kernel function,

Table 6.2: Accuracy of SVM classifier with respect to various kernel functions.

Accuracy	Kernel functions			
	Linear	Gaussian RBF	Polynomial (of degree 4)	Chi-square
Training	62%	91%	83%	86%
Testing	47%	87%	74%	79%

the values of (C, γ) are computed using loose and fine grid search methods. The accuracy of the SVM classifier for different value pairs of (C, γ) is depicted in Fig. 6.9.

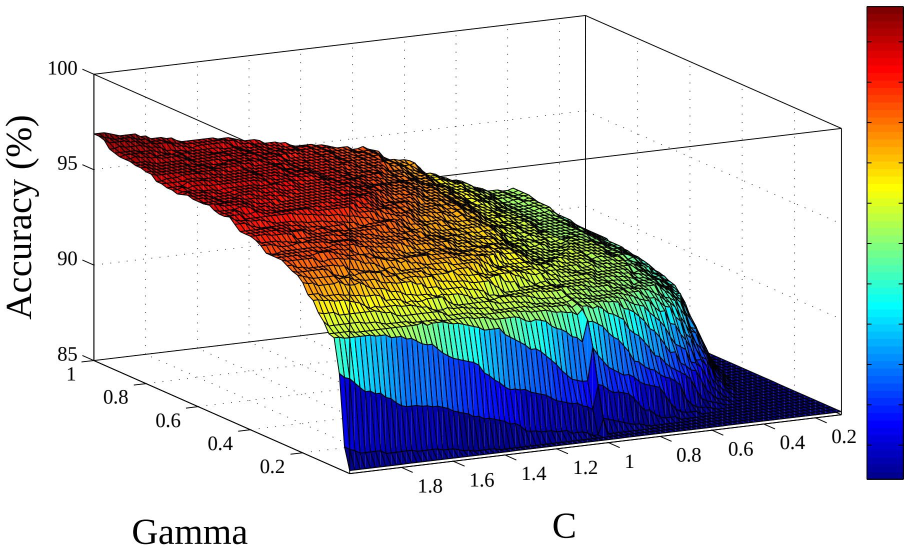


Figure 6.9: Accuracy of SVM classifier on different value pairs of (C, γ).

It can be inferred from the figure that the SVM classifier gives maximum accuracy on the pair value of $(1.88, 0.92)$. The SVM is then trained on the basis of these values to correctly classify the training values. It is to be noted here that the training accuracy of SVM classifier increases if the initial training proportion is large. This is evident from Fig. 6.10 which depicts the training accuracies of SVM classifier when different percentage of data chunks are used to train the classifier. However, the large percentage of training data should not be used to train the classifier as it may lead to the problem of over-fitting which hampers its generalization capabilities on the test data. Therefore, to avoid over-fitting problem while maximizing the accuracy of classifier, 70% of initial training data is used to train the classifier while

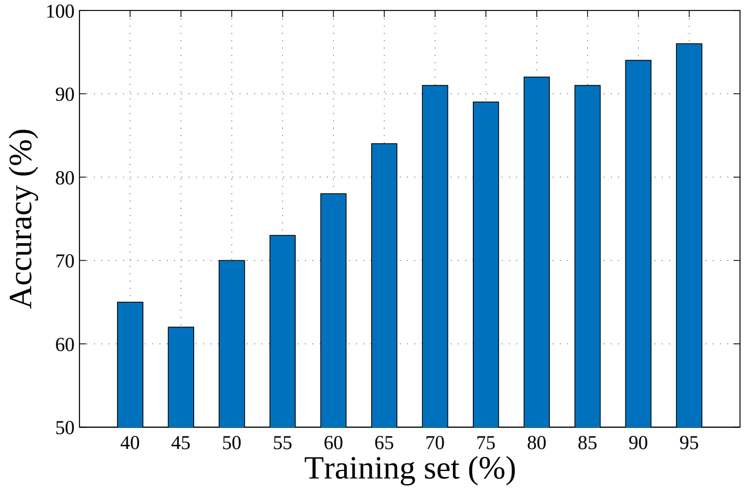


Figure 6.10: Accuracy of SVM classifier on different percentage of training set.

the remaining data is used to test its effectiveness. The results on this division of dataset for training and testing of SVM classifier on different fold cross-validation are given in Fig. 6.11. This figure illustrates that the maximum 10-fold cross-validation accuracy for the training set is 91% while the test set accuracy of the classifier comes out to be 87%.

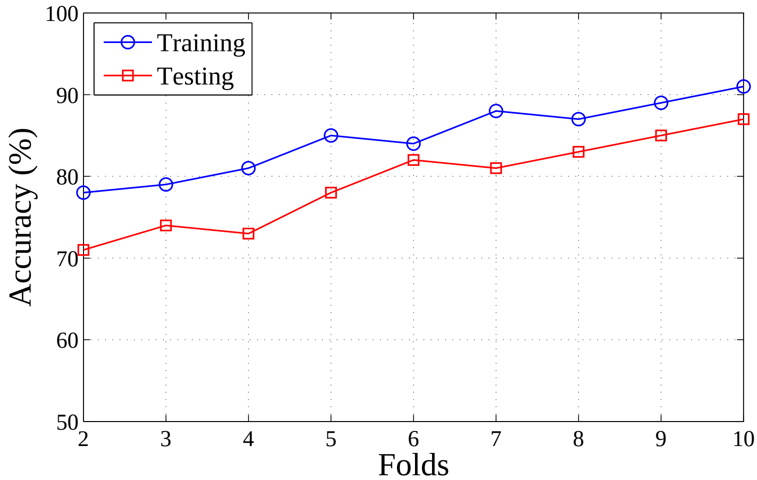


Figure 6.11: Cross-validation accuracy of SVM classifier.

Apart from the high accuracy exhibited by the SVM classifier, another advantage of using is that it is able to classify the users in a very less time which makes it scalable for even the large number of end-users. This is evident from Fig. 6.12 which shows the classification time taken by the SVM classifier to classify the users request received at a particular time instant. Moreover, this figure also highlights the benefit of using the tensor-based processing for storing data in the tensor form. It can be seen from Fig. 6.12 that the classification time taken for processing request from tensor data is much less than the requests that arrive directly to the SVM classifier. However, the classification time is machine dependent and this computational analysis has been carried out by using Intel Core i3-3110M CPU @ 2.40 GHz with 4GB of primary memory.

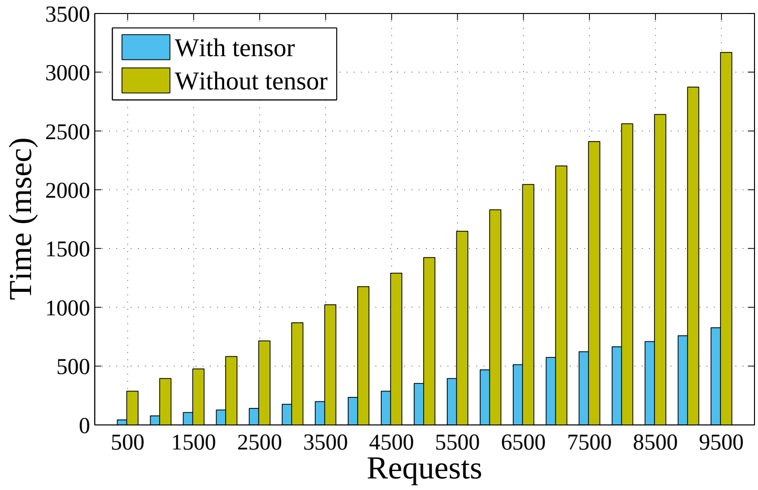


Figure 6.12: Classification time taken by the SVM classifier.

This SVM classifier is used to classify the end-users (i.e. residential loads and commercial loads) into three categories, i.e., overloaded, normal and underloaded. The results of this categorization for one time-slot of 15-minute duration for given number of residential and commercial loads are depicted in Figs. 6.13 and 6.14.

Based on this classification, the utility can take informed decisions so as to manage the demand response of overloaded and underloaded users to strike a balance between demand and supply in a smart city. Moreover, the utility companies can devise certain strategies to lure these overloaded and underloaded users to manage their load requirements and maintain a specific load profile which would optimally use power generation resources. However, for this purpose, the consumers need to be identified that can be benefited from these strategies which is achieved with the help of proposed SVM classifier.

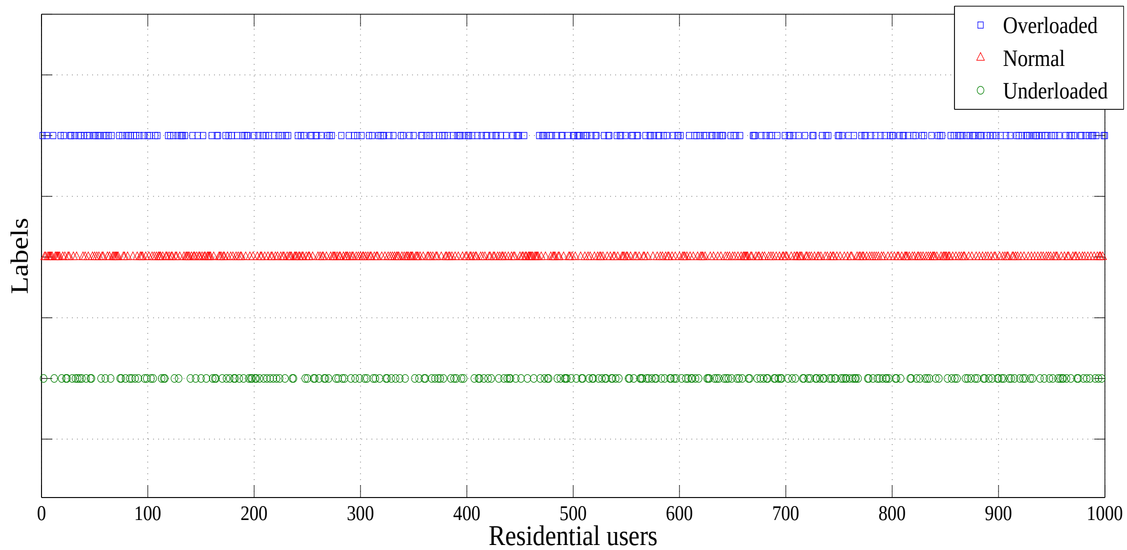


Figure 6.13: SVM classification of residential users.

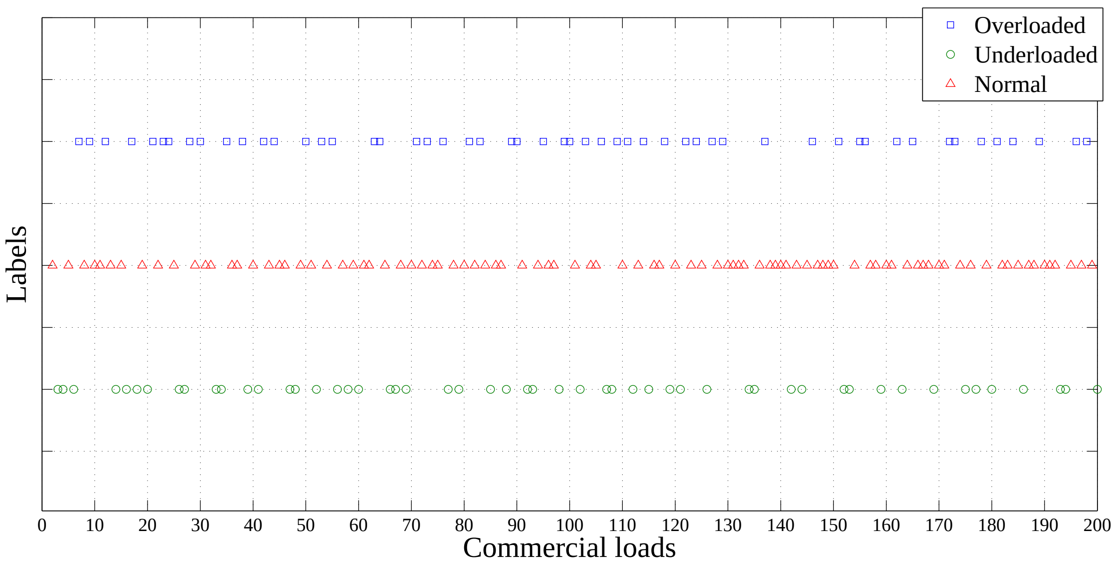


Figure 6.14: SVM classification of commercial users.

6.6 Summary

A tensor-based data processing technique in IoE environment has been presented in this chapter. To store the gathered data in a manageable form, its dimensionality is reduced by applying various tensor operations on it like matricization, vectorization and tensorization. The core data and projection matrices extracted out of original data which are later used to provide demand response services. For this purpose, the data is given as an input to the SVM classifier which assigns labels to various residential and commercial end-users in order to identify the overloaded and under-loaded users. The tensor-based processing reduces the data storage to 38% while giving the re-construction accuracy of 95.84%; while SVM classifier assigns the labels to users with the accuracy of 87% for the test set. The proposed tensor-based processing of data is advantageous and can be extended to provide other services in the smart cities as well such as-energy forecasting, electricity price prediction, outage detection, etc.

Chapter 7

Conclusion and Future scope

This chapter gives the concluding remarks on the research work related to applying data analytics in the smart grid environment for various purposes such as theft detection and managing the demand response. Moreover, it also discusses some open issues which can be taken as future works in these domains.

7.1 Conclusion

This thesis introduces the basic concepts related to smart grid, demand response, data analytics along with the need and impact of applying data analytics in the smart grid environment. Moreover, it also sheds the light on various challenges and constraints involved while performing data analytics in smart grid. In addition to it, the detailed literature review is performed in the domains of demand response, data analytics and energy management in the smart grid. Moreover, the relative comparison of existing approaches is also carried out to find their relative advantages and disadvantages in terms of various evaluation parameters. The main aim of this work is to address the research questions related to the applicability of data analytics in the smart grid environment for theft detection and demand response management. Moreover, the questions related to study the effect of load re-scheduling of the connected loads such as smart appliances and EVs, managing the demand response of the users without affecting user comfort and efficient handling of the collected data are also addressed. To answer these questions, four schemes have been developed on the basis of data analytics in this thesis pertaining to the theft detection in the smart grid environment, demand response management and efficient handling of the gathered data. The summary of each of these schemes is given as follows.

In the first scheme, a novel top-down approach has been presented using DT and SVM classifier to detect the malicious consumers that intentionally steal the electricity. The DT is used to predict the expected consumption in each smart home. This predicted consumption along with other features is provided as an input to the SVM classifier which is trained on the collected dataset. Once this classifier is trained, it is then used to classify the consumers as normal or fraud based upon their respective features. The results on the dataset prepared from Open Energy Information along with the U.S. Energy Information Administration depict that the proposed scheme identifies fraudulent consumers with an accuracy of 92.5% and has a false positive rate as low as 5.12%. This scheme also works well in a scenario where very less number of smart meters are deployed. Results also prove that the accuracy of the SVM classifier is improved when it works in conjunction with DT.

In the second scheme, the demand response of smart homes and PHEVs is managed in the energy network so as to maintain the overall load stability in smart grid at all the times. The SVM classifier has been used to identify the users using excess electricity consumption; and the load of such users is then curtailed with the help of a load balancing algorithm. If the residential loads are not sufficient to reduce the demand and supply gap, then the SVM classifier discovers the PHEVs whose charging rates can be regulated. The load profile of smart grid is then managed by altering the charging rates of such PHEVs according to the grid requirements. The simulation results of this scheme on PJM and Open Energy Information prove that the proposed scheme is effective in maintaining the overall load profile of smart grid by managing the demand response of smart homes and PHEV users. Moreover, the classification time taken by the SVM classifier is very less and its accuracy is very high; thereby making the overall scheme very fast and reliable.

The third scheme emphasizes on the issue of peak load reduction at smart grid on the basis of the data gathered from smart homes. For this purpose, various factors have been designed which help in taking efficient demand response decisions. These decisions are taken with the help of algorithms proposed from the consumer and utility perspectives. Additionally, an incentive scheme is also developed to increase the consumer participation in the proposed scheme and to compensate for comfort violations. The results obtained on the simulation of the proposed scheme depict that this scheme reduces the peak load on the grid by 14.84% using consumer-centric algorithm alone, while the same is 27.38% if both algorithms are used in tandem. Moreover, results also clearly reflect towards the dual benefit of the proposed scheme. Firstly, the proposed scheme effectively manages peak load demand in smart grid as depicted using two different scenarios. Secondly, the consumer is also benefited in terms of reduced electricity bills because of direct and indirect price savings.

In the fourth scheme, a tensor-based data processing technique is applied in smart grid environment to store the gathered data in a manageable form. For this purpose, the data dimensionality is reduced by applying various tensor operations on it like matricization, vectorization and tensorization. Using these operations, the core data and projection matrices are extracted out of original data which are later used to provide demand response services. To manage demand response, labels are assigned to various residential and commercial end-users in order to identify the overloaded and underloaded users with the help of the SVM classifier. The proposed tensor-based processing reduces the data storage to a great extent while giving very high re-construction accuracy. The results also depict that classification time taken by the SVM classifier in case of tensor-based processing is also less as compared to the scenario where requests are processed directly.

7.2 Future Scope

The future research directions in the field of data analytics in the smart grid environment can be focused on the following aspects.

- To incorporate the big data handling techniques for analyzing the rapidly growing data and manage the load requirements in the metropolitan cities.
- The robustness of the proposed schemes can be tested for fault tolerance, security and privacy preservation in real-time with increasing pace of data collection.
- A mechanism for unwanted data removal, before processing it, can be incorporated in the proposed schemes to improve their overall accuracies. However, the overhead generated for such a mechanism is required to be evaluated so as to make a trade-off between the overhead calculation and accuracy.
- The efficacy of the proposed schemes can be tested by adding information about other parameters that are directly or indirectly related to the smart grid.
- The effect of underlying communication architecture used like 4G and 5G can be studied on the performance of the proposed schemes.

Bibliography

[1] U.S. Energy Information Administration, Annual Energy Outlook 2016, Table A8. Electricity supply, disposition, prices, and emissions, Reference case: 2015. Available: http://www.eia.gov/outlooks/aeo/pdf/0383(2016).pdf.

[2] U.S. Energy Information Administration, Annual Energy Outlook 2016, Table A4. Residential sector key indicators and consumption, Reference case: April 2015. Available: https://www.eia.gov/outlooks/aeo/pdf/0383(2016).pdf.

[3] A. Cichocki, N. Lee, I. Oseledets, A.-H. Phan, Q. Zhao, D. P. Mandic *et al.*, "Tensor networks for dimensionality reduction and large-scale optimization: Part 1 low-rank tensor decompositions," *Foundations and Trends® in Machine Learning*, vol. 9, no. 4-5, pp. 249–429, 2016.

[4] G. Piro, I. Cianci, L. A. Grieco, G. Boggia, and P. Camarda, "Information centric services in smart cities," *Journal of Systems and Software*, vol. 88, pp. 169–188, 2014.

[5] Ministry of Urban Development, "Smart cities: Mission statement & guidelines," *Government of India*, 2015. [Online]. Available: http://smartcities.gov.in/upload/uploadfiles/files/SmartCityGuidelines(1).pdf.

[6] X. Yu, C. Cecati, T. Dillon, and M. G. Simoes, "The new frontier of smart grids," *IEEE Industrial Electronics Magazine*, vol. 5, no. 3, pp. 49–63, 2011.

[7] X. Fang, S. Misra, G. Xue, and D. Yang, "Smart grid-the new and improved power grid: A survey," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 4, pp. 944–980, 2012.

[8] N. Bui, A. P. Castellani, P. Casari, and M. Zorzi, "The internet of energy: a web-enabled smart grid system," *IEEE Network*, vol. 26, no. 4, 2012.

[9] P. Siano, "Demand response and smart grids - A survey," *Renewable and sustainable energy reviews*, vol. 30, pp. 461–478, 2014.

[10] M. H. Albadi and E. F. El-Saadany, "Demand response in electricity markets: An overview," in *IEEE Power Engineering Society General Meeting*, 2007, pp. 1–5.

[11] R. K. Chauhan, B. S. Rajpurohit, F. M. Gonzalez-Longatt, and S. N. Singh, "Intelligent energy management system for PV-battery-based microgrids in future DC homes," *International Journal of Emerging Electric Power Systems*, vol. 17, no. 3, pp. 339–350, 2016.

[12] S. Pannala, N. Padhy, and P. Agarwal, "Peak energy management using renewable integrated DC microgrid," *IEEE Transactions on Smart Grid*, 2017, DOI: 10.1109/TSG.2017.2675917.

[13] K. Kok, S. Karnouskos, D. Nestle, A. Dimeas, A. Weidlich, C. Warmer, P. Strauss, B. Buchholz, S. Drenkard, N. Hatziargyriou *et al.*, "Smart houses for a smart grid," in *20th International Conference and Exhibition on Electricity Distribution-Part 1*, 2009, pp. 1–4.

[14] S. Ge, P. Zeng, R. Lu, and K.-K. R. Choo, "FGDA: Fine-grained data analysis in privacy-preserving smart grid communications," *Peer-to-Peer Networking and Applications*, pp. 1–13, 2017.

[15] M. B. Naik, P. Kumar, and S. Majhi, "Small-scale solar plants coupled with smart public transport system and its coordination with the grid," *IET Electrical Systems in Transportation*, vol. 7, no. 2, pp. 135–144, 2016.

[16] J. K. Liu, W. Susilo, T. H. Yuen, M. H. Au, J. Fang, Z. L. Jiang, and J. Zhou, "Efficient privacy-preserving charging station reservation system for electric vehicles," *The Computer Journal*, vol. 59, no. 7, pp. 1040–1053, 2016.

[17] Global EV Outlook 2016, Available: https://www.iea.org/publications/freepublications/publication/Global_EV_Outlook_2016.pdf, [Accessed: Mar. 2017].

[18] A. Bilh, K. Naik, and R. El-Shatshat, "An adaptive charging algorithm for electric vehicles in smart grids," in *IEEE 81st Vehicular Technology Conference (VTC Spring)*, 2015, pp. 1–7.

[19] J. Tomić and W. Kempton, "Using fleets of electric-drive vehicles for grid support," *Journal of power sources*, vol. 168, no. 2, pp. 459–468, 2007.

[20] Z. Wang and S. Wang, "Grid power peak shaving and valley filling using vehicle-to-grid systems," *IEEE Transactions on power delivery*, vol. 28, no. 3, pp. 1822–1829, 2013.

[21] S. Das, Y. Ohba, M. Kanda, D. Famolari, and S. K. Das, "A key management framework for AMI networks in smart grid," *IEEE Communications Magazine*, vol. 50, no. 8, pp. 30–37, August 2012.

[22] A. Rose, "How big data is about to ignite smart grids worldwide," 2014. [Online]. Available: http://www.greenbiz.com/blog/2014/08/08/big-data-transform-smart-grids-worldwide.

[23] National Institute of Standards and Technology, "Strategic R&D opportunities for the smart grid," 2013. [Online]. Available:

http://www.nist.gov/smartgrid/upload/Final-Version-22-Mar-2013-Strategic-R-D-Opportunities-for-the-Smart-Grid.pdf.

[24] S. Laitner, M. T. McDonnell, and K. Ehrhardt-Martinez, "The energy efficiency and productivity benefits of smart appliances and ict-enabled networks: An initial assessment," *American Council for an Energy Efficient Economy.*

[25] C. L. Stimmel, *Big Data Analytics Strategies for the Smart Grid.* CRC Press, 2015.

[26] S. Chaudhuri and U. Dayal, "An overview of data warehousing and OLAP technology," *ACM Sigmod record*, vol. 26, no. 1, pp. 65–74, 1997.

[27] P. Mirowski, S. Chen, T. Kam Ho, and C.-N. Yu, "Demand forecasting in smart grids," *Bell Labs technical journal*, vol. 18, no. 4, pp. 135–158, 2014.

[28] J. Zhu, E. Zhuang, C. Ivanov, and Z. Yao, "A data-driven approach to interactive visualization of power systems," *IEEE Transactions on Power Systems*, vol. 26, no. 4, pp. 2539–2546, 2011.

[29] Z. Wu, T. Zhao, L. He, and X. Shen, "Smart grid meter analytics for revenue protection," in *International Conference on Power System Technology (POWERCON)*, 2014, pp. 782–787.

[30] General Electric, Available: https://www.ge.com/digital/products/asset-performance-management-for-power, [Accessed: May 2017].

[31] SIEMENS - Smart Metering, Available: http://w3.siemens.com/smartgrid/global/en/products-systems-solutions/smart-metering/Pages/overview.aspx, [Accessed: May 2017].

[32] Auto Grid, Available: http://www.auto-grid.com/solutions/, [Accessed: May 2016].

[33] Oracle Utilities Opower Peak Management, Available: https://www.oracle.com/industries/utilities/products/opower-peak-management-cloud-service/index.html, [Accessed: May 2017].

[34] OGE Energy Corp., Available: https://www.oge.com/wps/portal/oge/save-energy/silver-energy/, [Accessed: Sept. 2017].

[35] Con Edison, Available: https://www.smartgrid.gov/project/consolidated_edison_company_new_york_inc_smart_grid_deployment_project.html, [Accessed: Sept. 2017].

[36] R. Martin, "The installed base of smart meters will surpass 1 billion by 2022," *Navigant Research*, 2013. [Online]. Available: https://www.navigantresearch.com/newsroom/the-installed-base-of-smart-meters-will-surpass-1-billion-by-2022.

[37] D. J. Leeds, "The soft grid 2013-2020: Big data & utility analytics for smart grid," 2012. [Online]. Available: http://www.greentechmedia.com/research/report/the-soft-grid-2013.

[38] PR Newswire, Available: http://www.prnewswire.com/news-releases/world-loses-893-billion-to-electricity-theft-annually-587-billion-in-emerging-markets-300006515.html, [Accessed: Jul. 2015].

[39] P. Jokar, N. Arianpoo, and V. Leung, "Intrusion detection in advanced metering infrastructure based on consumption pattern," in *IEEE International Conference on Communications (ICC)*, 2013, pp. 4472–4476.

[40] W. Han, L. Zhang, and J. Liu, "Demand response model for characteristics analysis of electricity consumers," in *IEEE Innovative Smart Grid Technologies-Asia (ISGT Asia)*, 2012, pp. 1–5.

[41] K. Wang, C. Xu, Y. Zhang, S. Guo, and A. Zomaya, "Robust big data analytics for electricity price forecasting in the smart grid," *IEEE Transactions on Big Data*, 2017, DOI: 10.1109/TBDATA.2017.2723563.

[42] S. Haben, C. Singleton, and P. Grindrod, "Analysis and clustering of residential customers energy behavioral demand using smart meter data," *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 136–144, 2016.

[43] D. Alahakoon and X. Yu, "Smart electricity meter data intelligence for future energy systems: A survey," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 1, pp. 425–436, 2016.

[44] N. Yamaguchi, J. Han, G. Ghatikar, S. Kiliccote, M. A. Piette, and H. Asano, "Regression models for demand reduction based on cluster analysis of load profiles," in *IEEE PES/IAS Conference on Sustainable Alternative Energy (SAE)*, 2009, pp. 1–7.

[45] Y. Zhang, W. Chen, R. Xu, and J. Black, "A cluster-based method for calculating baselines for residential loads," *IEEE Transactions on Smart Grid*, vol. 7, no. 5, pp. 2368–2377, 2016.

[46] D. Zhou, M. Balandat, and C. Tomlin, "Residential demand response targeting using machine learning with observational data," in *IEEE 55th Conference on Decision and Control (CDC)*, 2016, pp. 6663–6668.

[47] L. Hatton, P. Charpentier, and E. Matzner-Løber, "Statistical estimation of the residential baseline," *IEEE Transactions on Power Systems*, vol. 31, no. 3, pp. 1752–1759, 2016.

[48] C.-K. Woo, I. Horowitz, and I. M. Sulyma, "Relative kw response to residential time-varying pricing in british columbia," *IEEE Transactions on Smart Grid*, vol. 4, no. 4, pp. 1852–1860, 2013.

[49] E. Klaassen, J. Frunt, and H. Slootweg, "Analyzing the impact of smart grids on load profiles using multiple regression," in *50th International Universities Power Engineering Conference (UPEC)*, 2015, pp. 1–6.

[50] J. Jazaeri, T. Alpcan, R. Gordon, M. Brandao, T. Hoban, and C. Seeling, "Baseline methodologies for small scale residential demand response," in *IEEE Innovative Smart Grid Technologies-Asia (ISGT-Asia)*, 2016, pp. 747–752.

[51] Y. Wang, Q. Chen, C. Kang, M. Zhang, K. Wang, and Y. Zhao, "Load profiling and its application to demand response: A review," *Tsinghua Science and Technology*, vol. 20, no. 2, pp. 117–129, 2015.

[52] D. Zhang, S. Li, M. Sun, and Z. O'Neill, "An optimal and learning-based demand response and home energy management system," *IEEE Transactions on Smart Grid*, vol. 7, no. 4, pp. 1790–1801, 2016.

[53] S. McLaughlin, B. Holbert, A. Q. Fawaz, R. Berthier, and S. Zonouz, "A multi-sensor energy theft detection framework for advanced metering infrastructures," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 7, pp. 1319–1330, 2013.

[54] G. Sreenivasan, *POWER THEFT: Educates and sensities people about the menace of power theft.* PHI Learning Pvt. Ltd., 2011.

[55] Y. Guo, C.-W. Ten, and P. Jirutitijaroen, "Online data validation for distribution operations against cybertampering," *Power Systems, IEEE Transactions on*, vol. 29, no. 2, pp. 550–560, 2014.

[56] A. Giani, E. Bitar, M. Garcia, M. McQueen, P. Khargonekar, and K. Poolla, "Smart grid data integrity attacks: characterizations and countermeasures," in *IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Oct 2011, pp. 232–237.

[57] C. Muniz, M. M. B. R. Vellasco, R. Tanscheit, and K. Figueiredo, “A neuro-fuzzy system for fraud detection in electricity distribution.” in *Proceedings of the Joint International Fuzzy Systems Association World Congress and 2009 European Society of Fuzzy Logic and Technology Conference, Lisbon, Portugal*, Jul. 2009, pp. 1096–1101.

[58] B. C. Costa, B. L. Alberto, A. M. Portela, W. Maduro, and E. O. Eler, “Fraud detection in electric power distribution networks using an ANN based knowledge-discovery process,” *International Journal of Artificial Intelligence and Applications*, vol. 4, no. 6, pp. 17–23, 2013.

[59] E. Angelos, O. Saavedra, O. Cortes, and A. D. Souza, “Detection and identification of abnormalities in customer consumptions in power distribution systems,” *IEEE Transactions on Power Delivery*, vol. 26, no. 4, pp. 2436–2442, Oct. 2011.

[60] J. Cabral, “Fraud detection in electrical energy consumers using rough sets,” in *IEEE International Conference on Systems, Man and Cybernetics*, vol. 4, Oct. 2004, pp. 3625–3629.

[61] J. Nagi, A. M. Mohammad, K. S. Yap, S. K. Tiong, and S. K. Ahmed, “Non-technical loss analysis for detection of electricity theft using support vector machines,” in *2nd IEEE International Power and Energy Conference (PECon)*, 2008, pp. 907–912.

[62] S. S. S. R. Depuru, L. Wang, and V. Devabhaktuni, “Support vector machine based data classification for detection of electricity theft,” in *IEEE/PES Power Systems Conference and Exposition (PSCE)*, 2011, pp. 1–8.

[63] S. S. S. R. Depuru, L. Wang, V. Devabhaktuni, and P. Nelapati, “A hybrid neural network model and encoding technique for enhanced classification of energy consumption data,” in *IEEE Power and Energy Society General Meeting*, 2011, pp. 1–8.

[64] J. Nagi, K. S. Yap, S. K. Tiong, S. Ahmed, and F. Nagi, “Improving SVM-based nontechnical loss detection in power utility using the fuzzy inference system,” *IEEE Transactions on Power Delivery*, vol. 26, no. 2, pp. 1284–1285, Apr. 2011.

[65] R. K. Chauhan, B. S. Rajpurohit, L. Wang, F. G. Longatt, and S. Singh, “Real time energy management system for smart buildings to minimize the electricity bill,” *International Journal of Emerging Electric Power Systems*, vol. 18, no. 3, pp. 1–15, 2017.

[66] R. Zhou, Z. Li, C. Wu, and M. Chen, “Demand response in smart grids: A randomized auction approach,” *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 12, pp. 2540–2553, 2015.

[67] A. Safdarian, M. Fotuhi-Firuzabad, and M. Lehtonen, “Optimal residential load management in smart grids: a decentralized framework,” *IEEE Transactions on Smart Grid*, vol. 7, no. 4, pp. 1836–1845, 2016.

[68] N. Lu, “An evaluation of the hvac load potential for providing load balancing service,” *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1263–1270, 2012.

[69] G. T. Costanzo, G. Zhu, M. F. Anjos, and G. Savard, “A system architecture for autonomous demand side load management in smart buildings,” *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 2157–2165, 2012.

[70] T.-H. Chang, M. Alizadeh, and A. Scaglione, “Real-time power balancing via decentralized coordinated home energy scheduling,” *IEEE Transactions on Smart Grid*, vol. 4, no. 3, pp. 1490–1504, 2013.

[71] P. Chavali, P. Yang, and A. Nehorai, “A distributed algorithm of appliance scheduling for home energy management system,” *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 282–290, 2014.

[72] M. Pipattanasomporn, M. Kuzlu, and S. Rahman, "An algorithm for intelligent home energy management and demand response analysis," *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 2166–2173, 2012.

[73] Y. Ozturk, D. Senthilkumar, S. Kumar, and G. Lee, "An intelligent home energy management system to improve demand response," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 694–701, 2013.

[74] A. Bourass, S. Cherkaoui, and L. Khoukhi, "Intelligent route guidance for electric vehicles in the smart grid," in *IEEE Global Communications Conference (GLOBECOM)*, 2017, pp. 1–7.

[75] Z. Fan, "A distributed demand response algorithm and its application to phev charging in smart grids," *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1280–1290, 2012.

[76] D. Ban, G. Michailidis, and M. Devetsikiotis, "Demand response control for phev charging stations by dynamic price adjustments," in *IEEE PES Innovative Smart Grid Technologies (ISGT)*, 2012, pp. 1–8.

[77] Y. Yao and D. W. Gao, "Charging load from large-scale plug-in hybrid electric vehicles: Impact and optimization," in *IEEE PES Innovative Smart Grid Technologies (ISGT)*, 2013, pp. 1–6.

[78] R. K. Jain, K. M. Smith, P. J. Culligan, and J. E. Taylor, "Forecasting energy consumption of multi-family residential buildings using support vector regression: Investigating the impact of temporal and spatial monitoring granularity on performance accuracy," *Applied Energy*, vol. 123, pp. 168–178, 2014.

[79] R. E. Edwards, J. New, and L. E. Parker, "Predicting future hourly residential electrical consumption: A machine learning case study," *Energy and Buildings*, vol. 49, pp. 591–603, 2012.

[80] H. Chitsaz, H. Shaker, H. Zareipour, D. Wood, and N. Amjady, "Short-term electricity load forecasting of buildings in microgrids," *Energy and Buildings*, vol. 99, pp. 50–60, 2015.

[81] S. K. Nayak, N. Sahoo, and G. Panda, "Demand side management of residential loads in a smart grid using 2d particle swarm optimization technique," in *IEEE Power, Communication and Information Technology Conference (PCITC)*, 2015, pp. 201–206.

[82] M. Shafie-khah, E. Heydarian-Forushani, G. J. Osório, F. A. Gil, J. Aghaei, M. Barani, and J. P. Catalão, "Optimal behavior of electric vehicle parking lots as demand response aggregation agents," *IEEE Transactions on Smart Grid*, vol. 7, no. 6, pp. 2654–2665, 2016.

[83] A. Soroudi, P. Siano, and A. Keane, "Optimal DR and ESS scheduling for distribution losses payments minimization under electricity price uncertainty," *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 261–272, 2016.

[84] G. Escrivá-Escrivá, "Basic actions to improve energy efficiency in commercial buildings in operation," *Energy and Buildings*, vol. 43, no. 11, pp. 3106–3111, 2011.

[85] Z. Zhou, F. Zhao, and J. Wang, "Agent-based electricity market simulation with demand response from commercial buildings," *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 580–588, 2011.

[86] B. Chai, A. Costa, S. D. Ahipasaoglu, C. Yuen, and Z. Yang, "Optimal meeting scheduling in smart commercial building for energy cost reduction," *IEEE Transactions on Smart Grid*, 2016, DOI: 10.1109/TSG.2016.2625313.

[87] T. Logenthiran, D. Srinivasan, and T. Z. Shun, "Demand side management in smart grid using heuristic optimization," *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1244–1252, 2012.

[88] S. Mohagheghi and N. Raji, "Dynamic demand response: A solution for improved energy efficiency for industrial customers," *IEEE Industry Applications Magazine*, vol. 21, no. 2, pp. 54–62, 2015.

[89] M. Choobineh and S. Mohagheghi, "Optimal energy management in an industrial plant using on-site generation and demand scheduling," in *IEEE Industry Applications Society Annual Meeting*, 2015, pp. 1–8.

[90] H. Hao, C. D. Corbin, K. Kalsi, and R. G. Pratt, "Transactive control of commercial buildings for demand response," *IEEE Transactions on Power Systems*, vol. 32, no. 1, pp. 774–783, 2017.

[91] G. Goddard, J. Klose, and S. Backhaus, "Model development and identification for fast demand response in commercial hvac systems," *IEEE Transactions on Smart Grid*, vol. 5, no. 4, pp. 2084–2092, 2014.

[92] K. Sakurama and M. Miura, "Communication-based decentralized demand response for smart microgrids," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 5192–5202, 2017.

[93] H. Huang, Y. Cai, H. Xu, and H. Yu, "A multi-agent minority-game based demand-response management of smart buildings towards peak load reduction," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 36, no. 4, pp. 573–585, 2017.

[94] F. Ruelens, B. J. Claessens, S. Vandael, B. D. Schutter, R. Babuška, and R. Belmans, "Residential demand response of thermostatically controlled loads using batch reinforcement learning," *IEEE Transactions on Smart Grid*, vol. 8, no. 5, pp. 2149–2159, 2017.

[95] S. A. Pourmousavi, S. N. Patrick, and M. H. Nehrir, "Real-time demand response through aggregate electric water heaters for load shifting and balancing

wind generation," *IEEE Transactions on Smart Grid*, vol. 5, no. 2, pp. 769–778, 2014.

[96] C. Vivekananthan, Y. Mishra, G. Ledwich, and F. Li, "Demand response for residential appliances via customer reward scheme," *IEEE Transactions on Smart Grid*, vol. 5, no. 2, pp. 809–820, 2014.

[97] S. Althaher, P. Mancarella, and J. Mutale, "Automated demand response from home energy management system under dynamic pricing and power and comfort constraints," *IEEE Transactions on Smart Grid*, vol. 6, no. 4, pp. 1874–1883, 2015.

[98] W. Tushar, B. Chai, C. Yuen, S. Huang, D. B. Smith, H. V. Poor, and Z. Yang, "Energy storage sharing in smart grid: A modified auction-based approach," *IEEE Transactions on Smart Grid*, vol. 7, no. 3, pp. 1462–1475, 2016.

[99] W. Tushar, J. A. Zhang, C. Yuen, D. B. Smith, and N. U. Hassan, "Management of renewable energy for a shared facility controller in smart grid," *IEEE Access*, vol. 4, pp. 4269–4281, 2016.

[100] J. Byun, I. Hong, and S. Park, "Intelligent cloud home energy management system using household appliance priority based scheduling based on prediction of renewable energy capability," *IEEE Transactions on Consumer Electronics*, vol. 58, no. 4, 2012.

[101] Z. Zhao, W. C. Lee, Y. Shin, and K.-B. Song, "An optimal power scheduling method for demand response in home energy management system," *IEEE Transactions on Smart Grid*, vol. 4, no. 3, pp. 1391–1400, 2013.

[102] Z. Wang, C. Gu, F. Li, P. Bale, and H. Sun, "Active demand response using shared energy storage for household energy management," *IEEE Transactions on Smart Grid*, vol. 4, no. 4, pp. 1888–1897, 2013.

[103] M. Shakeri, M. Shayestegan, H. Abunima, S. S. Reza, M. Akhtaruzzaman, A. Alamoud, K. Sopian, and N. Amin, "An intelligent system architecture in home energy management systems (HEMS) for efficient demand response in smart grid," *Energy and Buildings*, vol. 138, pp. 154–164, 2017.

[104] X. Chen, T. Wei, and S. Hu, "Uncertainty-aware household appliance scheduling considering dynamic electricity pricing in smart home," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 932–941, 2013.

[105] N. G. Paterakis, O. Erdinc, A. G. Bakirtzis, and J. P. Catalão, "Optimal household appliances scheduling under day-ahead pricing and load-shaping demand response strategies," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 6, pp. 1509–1519, 2015.

[106] H.-C. Jo, S. Kim, and S.-K. Joo, "Smart heating and air conditioning scheduling method incorporating customer convenience for home energy management system," *IEEE Transactions on Consumer Electronics*, vol. 59, no. 2, pp. 316–322, 2013.

[107] C. Chen, J. Wang, and S. Kishore, "A distributed direct load control approach for large-scale residential demand response," *IEEE Transactions on Power Systems*, vol. 29, no. 5, pp. 2219–2228, 2014.

[108] J. Han, C.-S. Choi, W.-K. Park, I. Lee, and S.-H. Kim, "Smart home energy management system including renewable energy based on ZigBee and PLC," *IEEE Transactions on Consumer Electronics*, vol. 60, no. 2, pp. 198–202, 2014.

[109] A. Anvari-Moghaddam, H. Monsef, and A. Rahimi-Kian, "Optimal smart home energy management considering energy saving and a comfortable lifestyle," *IEEE Transactions on Smart Grid*, vol. 6, no. 1, pp. 324–332, 2015.

[110] A. A. Munshi and A.-R. M. Yasser, "Big data framework for analytics in smart grids," *Electric Power Systems Research*, vol. 151, pp. 369–380, 2017.

[111] K. Zhou, C. Fu, and S. Yang, "Big data driven smart energy management: From big data to big insights," *Renewable and Sustainable Energy Reviews*, vol. 56, pp. 215–225, 2016.

[112] P. D. Diamantoulakis, V. M. Kapinas, and G. K. Karagiannidis, "Big data analytics for dynamic energy management in smart grids," *Big Data Research*, vol. 2, no. 3, pp. 94–101, 2015.

[113] R. Silipo, I. Adae, A. Hart, and M. Berthold, "Seven techniques for dimensionality reduction," *KNIME*, 2014, Available: https://www-cdn.knime.com/sites/default/files/inline-images/knime_seventechniquesdatadimreduction.pdf.

[114] L. T. Yang, L. Kuang, J. Chen, F. Hao, and C. Luo, "A holistic approach to distributed dimensionality reduction of big data," *IEEE Transactions on Cloud Computing*, 2015, DOI: 10.1109/TCC.2015.2449855.

[115] L. Zhang, L. Zhang, D. Tao, and X. Huang, "Tensor discriminative locality alignment for hyperspectral image spectral–spatial feature extraction," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 1, pp. 242–256, 2013.

[116] J. Sun, D. Tao, S. Papadimitriou, P. S. Yu, and C. Faloutsos, "Incremental tensor analysis: Theory and applications," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 2, no. 3, p. 11, 2008.

[117] Y. Zhu, J. He, and R. D. Lawrence, "A general framework for predictive tensor modeling with domain knowledge," *Data Mining and Knowledge Discovery*, vol. 29, no. 6, pp. 1709–1732, 2015.

[118] L. Kuang, F. Hao, L. T. Yang, M. Lin, C. Luo, and G. Min, "A tensor-based approach for big data representation and dimensionality reduction," *IEEE transactions on emerging topics in computing*, vol. 2, no. 3, pp. 280–291, 2014.

[119] M. Chen, S. Mao, and Y. Liu, "Big data: A survey," *Mobile Networks and Applications*, vol. 19, no. 2, pp. 171–209, 2014.

[120] L. Wei, H. Zhu, Z. Cao, X. Dong, W. Jia, Y. Chen, and A. V. Vasilakos, "Security and privacy for storage and computation in cloud computing," *Information Sciences*, vol. 258, pp. 371–386, 2014.

[121] J. R. Quinlan, *C4.5: programs for machine learning.* Elsevier, 2014.

[122] S. Ruggieri, "Efficient C4.5 [classification algorithm]," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 2, pp. 438–444, Mar. 2002.

[123] J. R. Quinlan, "Induction of decision trees," *Machine learning*, vol. 1, no. 1, pp. 81–106, 1986.

[124] M. Kantardzic, "Decision trees and decision rules," *Data Mining: Concepts, Models, Methods, and Algorithms, Second Edition*, pp. 169–198, 2012.

[125] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 27:1–27:27, 2011.

[126] N. Shrivastava, A. Khosravi, and B. Panigrahi, "Prediction interval estimation of electricity prices using PSO-tuned support vector machines," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 2, pp. 322–331, Apr. 2015.

[127] C. Souza, "Kernel functions for machine learning applications," [Accessed: July 2015]. [Online]. Available: http://crsouza.com/2010/03/kernel-functions-for-machine-learning-applications/

[128] C. W. Hsu, C. C. Chang, and C. J. Lin, "A practical guide to support vector classification," 2003.

[129] Open Energy Information, Available: http://en.openei.org/datasets/dataset/commercial-and-residential-hourly-load-profiles-for-all-tmy3-locations-in-the-united-states, [Accessed: Jul. 2015].

[130] AccuWeather Incorporation, Available: http://www.accuweather.com/en/us/united-states-weather, [Accessed: Jul. 2016].

[131] US Energy Information Administration, Available: http://www.eia.gov/consumption/residential/data/2009/index.cfm?view=consumption, [Accessed: Jul. 2015].

[132] K. Kaur, A. Dua, A. Jindal, N. Kumar, M. Singh, and A. Vinel, "A novel resource reservation scheme for mobile phevs in v2g environment using game theoretical approach," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 12, pp. 5653–5666, 2015.

[133] P. Jokar, N. Arianpoo, and V. C. Leung, "Electricity theft detection in ami using customers consumption patterns," *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 216–226, 2016.

[134] N. A. Shrivastava, A. Khosravi, and B. K. Panigrahi, "Prediction interval estimation of electricity prices using pso-tuned support vector machines," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 2, pp. 322–331, 2015.

[135] J. C. A. Anton, P. J. G. Nieto, C. B. Viejo, and J. A. V. Vilán, "Support vector machines used to estimate the battery state of charge," *IEEE Transactions on Power Electronics*, vol. 28, no. 12, pp. 5919–5926, 2013.

[136] V. Klass, M. Behm, and G. Lindbergh, "A support vector machine-based state-of-health estimation method for lithium-ion batteries under electric vehicle operation," *Journal of Power Sources*, vol. 270, pp. 262–272, 2014.

[137] J. Remmlinger, M. Buchholz, M. Meiler, P. Bernreuter, and K. Dietmayer, "State-of-health monitoring of lithium-ion batteries in electric vehicles by on-

board internal resistance estimation," *Journal of Power Sources*, vol. 196, no. 12, pp. 5357–5363, 2011.

[138] V. Klass, M. Behm, and G. Lindbergh, "Evaluating real-life performance of lithium-ion battery packs in electric vehicles," *Journal of The Electrochemical Society*, vol. 159, no. 11, pp. A1856–A1860, 2012.

[139] Metered Load Data, PJM., Available: http://www.pjm.com/markets-and-operations/ops-analysis/historical-load-data.aspx, [Accessed: Oct. 2017].

[140] S. Kishore and L. V. Snyder, "Control mechanisms for residential electricity demand in smartgrids," in *IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2010, pp. 443–448.

[141] A. Abdallah and X. Shen, "Lightweight security and privacy preserving scheme for smart grid customer-side networks," *IEEE Transactions on Smart Grid*, vol. 8, no. 3, pp. 1064–1074, 2017.

[142] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G. P. Hancke, "Smart grid and smart homes: key players and pilot projects," *IEEE Industrial Electronics Magazine*, vol. 6, no. 4, pp. 18–34, 2012.

[143] K. Sato, A. Moody, K. Mohror, T. Gamblin, B. R. d. Supinski, N. Maruyama, and S. Matsuoka, "FMI: Fault tolerant messaging interface for fast and transparent recovery," in *2014 IEEE 28th International Parallel and Distributed Processing Symposium*, May 2014, pp. 1225–1234.

[144] R. Yin, P. Xu, M. A. Piette, and S. Kiliccote, "Study on auto-DR and pre-cooling of commercial buildings with thermal mass in california," *Energy and Buildings*, vol. 42, no. 7, pp. 967–975, 2010.

[145] U.S. Energy Information Administration, Annual Energy Outlook 2016, Table A4. Residential sector key indicators and consumption, Reference case: 2015. Available: http://www.eia.gov/outlooks/aeo/pdf/0383(2016).pdf.

[146] Electricity Local, Available: https://www.electricitylocal.com/resources/statistics/, States: AK, NY, TX, CA, WI [Accessed: Nov. 2016].

[147] Daily Generation Capacity, PJM., Available: http://dataminer2.pjm.com/feed/day_gen_capacity, [Accessed: Oct. 2017].

[148] R. K. Tripathi, S. Dhuli, Y. N. Singh, and N. K. Verma, "Analysis of weights for optimal positioning of base station in a wireless sensor network," in *Twentieth National Conference on Communications (NCC)*, Feb 2014, pp. 1–4.