Machine Learning Algorithms and Applications for Sustainable Smart Grid

Di Wu

Department of Electrical & Computer Engineering
McGill University
Montreal, Canada

July 2018

A thesis submitted to McGill University in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Abstract

Smart grid is a complex electrical power network comprising different subsystems with a level of automation enabling the use of renewable energy while maintaining the grid stability and affordability of the energy. With the increasing attention on environment protection and development of sensors, communication, and computation tools, the smart grid concept has gained a fast development in recent years. It could significantly improve energy efficiency, allow deep decarbonization and protect the environment. Machine learning is of essential importance to enable intelligent power systems.

In this thesis, we use three pieces of work to demonstrate how the smart grid can benefit from machine learning algorithms. First, we note that workplace electric vehicle (EV) charging is now supported by more and more companies to encourage EV adoption which is environmentally friendly. In the meantime, renewable energies are becoming an important power source. We propose to address the challenges of energy management in office buildings integrated with photovoltaic (PV) systems and workplace EV charging with a stochastic programming framework. Two computationally efficient control algorithms, Stochastic Programming and Load forecasting for Energy management with Two stages (SPLET) and Sample Average Approximation based SPLET (SAA_SPLET) are proposed.

Secondly, accurate electricity load forecasting is of crucial importance for power system operation and smart grid energy management. Multiple kernel learning (MKL) is suitable for electricity load forecasting, because this type of method provides more flexibility than traditional kernel methods. However, conventional MKL methods usually lead to complex optimization problems. At the scale of residential homes, another important aspect of this application is that there may be very little data available to train a reliable forecasting model for a new home, while at the same time we may have prior knowledge learned from other homes. In particular, we first adopt boosting to learn an ensemble of multiple kernel regressors, and then we further extend this framework to the context of transfer learning when limited data is available for target homes.

Finally, we aim to tackle home energy management without knowing the system dynamics. We propose to formalize home energy management, including buying energy from or selling energy back to the power grid and EV charging scheduling as a Markov Decision Process (MDP) and propose two model-free reinforcement learning based control algorithms to address it. The objective for the proposed algorithms is to minimize the long-term operating cost. Simulation results are presented with real-world data and show that the proposed algorithms can significantly reduce the electricity cost as well as peak power consumptions of the home.

Abrégé

Le réseau intelligent est un réseau d'énergie électrique complexe comprenant différents sous-systèmes avec un niveau d'automatisation permettant l'utilisation d'énergie renouvelable tout en maintenant la stabilité du réseau et l'abordabilité de l'énergie. Avec l'attention croissante portée à la protection de l'environnement et au développement de capteurs, d'outils de communication et de calcul, le concept de réseau intelligent a connu un développement rapide ces dernières années. Cela pourrait améliorer considérablement l'efficacité énergétique, permettre une décarbonation profonde et protéger l'environnement. L'apprentissage automatique est essentiel pour permettre des systèmes d'alimentation intelligents.

Dans cette thèse, nous utilisons trois travaux pour démontrer comment le réseau intelligente peut bénèficier des algorithmes d'apprentissage automatique. Premièrement, nous notons que la recharge des véhicules électriques (VÉ) sur le lieu de travail est maintenant fournie par de plus en plus d'entreprises pour encourager l'adoption des VÉ qui soit respectueux de l'environnement. Entre-temps, les énergies renouvelables deviennent une source d'énergie importante. Nous proposons de relever les défis de la gestion de l'énergie dans les immeubles de bureaux équipés de systèmes photovoltaïques (PV) de bornes de recharge de VÉ sur le lieu de travail avec un cadre de programmation stochastique. Deux algorithmes de contrôle, calcul stochastique et prévision de charge pour la gestion de l'énergie avec deux étapes (SPLET) et SPLET (SAA_SPLET) basé sur la moyenne approximative d'échantillonnage sont proposés.

Deuxièmement, une prévision précise de la charge électrique est d'une importance cruciale pour le fonctionnement du système électrique et la gestion intelligente de l'énergie du réseau. L'apprentissage par noyaux multiples(MKL) convient à la prévision de la charge électrique, car ce type de méthode offre plus de flexibilité que les méthodes traditionnelles du noyau. Cependant, les méthodes MKL classiques conduisent généralement à des problèmes d'optimisation complexes. À l'échelle des maisons résidentielles, un autre aspect important de cette application est qu'il peut y avoir très peu de données disponibles pour former un modèle de prévision fiable pour une nouvelle maison, alors que nous pouvons en même temps avoir des connaissances antérieures apprises d'autres foyers. En particulier, nous adoptons d'abord le boosting pour apprendre un ensemble de régresseurs de noyaux multiples, puis nous étendons ce cadre au contexte de l'apprentissage par transfert lorsque des données limitées sont disponibles pour les foyers cibles.

Enfin, nous visons à gérer la gestion de l'énergie domestique sans connaître la dynamique du système. Nous proposons de formaliser la gestion de l'énergie domestique, y compris l'achat d'énergie ou la revente d'énergie au réseau électrique et l'ordonnancement de charge des VÉ comme un processus de décision de Markov (MDP) et proposons deux algorithmes d'apprentissage de renforcement libre pour s'y attaquer. L'objectif des algorithmes proposés est de minimiser les coûts d'exploitation à long terme. Les résultats de la simulation sont présentés avec des données du monde réel et montrent que les algorithmes proposés peuvent réduire considérablement le coût de l'électricité ainsi que les consommations d'énergie de pointe.

Acknowledgments

The time spent at McGill as a Ph.D. student reformulates me in several aspects, I am so grateful that I could have this opportunity to study here and work with so many talented people. I would like to express my deepest thanks to all the people who had and have been helping me during the completion of my Ph.D. program.

Firstly, I want to express my most sincere appreciation and deepest respect to my supervisors: Professor Benoit Boulet and Professor Doina Precup, who provide me with so much dedicated support: academically, personally, and financially. I learned a lot from you. The discussions with you, not only bring me to the academic research frontier but also help me to learn how to define a problem, solve the problem and present the results. Meanwhile, I also learned a lot in managing life in an efficient way. Without your patience and detailed guidance, I will not have such a fruitful and enjoyable life at McGill. You are the models that I should follow all my life. Thanks!

Many thanks to Professor Haibo Zeng and Professor Boon-Teck Ooi, who serve in my supervision committee. The discussions with you helped me a lot in understanding the smart grid problems. I would like to thank Professor Haibo Zeng for guiding me finishing the first research project in my Ph.D. program. I sincerely appreciate Professor Boon-Teck Ooi for helping understanding some fundamental concepts of power systems.

I am especially grateful for the insightful discussions I had with Xiaoyu Wang from Ottawa University, Chao Lu from Tsinghua University, Boyu Wang from Princeton University, Guillaume Rabusseau from McGill University and Vincent Franc̀ois-Lavet from McGill University. Your guidance and feedback on my research greatly helped me to formulate the problems in a more comprehensive way. I learned a lot of knowledge and interesting things from you. I would also like to express my sincere appreciation for professors in the Electrical and Computer Engineering department as well as professors in Computer Science department, for the instructions and feedbacks during the courses, seminars and talks.

I would like to express my appreciations for my colleagues in McGill University. I want to thank Yingxuan Duan, Hossein Vahid Alizadeh, Mir Saman Rahimi, Chuansheng Dong, Hangcheng Zhu, Fanxin Kong, Zaid Albayati, Hassan Mozakari, FeiZhang, Qiushi Cui. Thanks for the discussions on my academic research as well as the support during the hardship in these years.

I want to say thanks to my parents Lei Zhao and Liya Yang, who always try their

best to support me and try to help me get a better understanding for the world, the life, the love, and the responsibility. I would also want to express my thanks for my grand parents. Thanks, you gave me a wonderful childhood in a beautiful rural village. Finally, many thanks to my wife Shan Jiang, you are such a nice, kind and self-disciplined person. Thanks a lot for your support and understanding during the journey of completing my Ph.D. program. I am so lucky to meet you.

The world is changing so fast now, and a lot of things have changed in last five years. Again, I am grateful that I could come here, know these people and work with these people.

Acronyms

Acronyms	Descriptions
ANN	Artificial neural network
ASAP	As soon as possible
BSOC	Battery State of Charge
DQN	Deep Q Network
DQNEMS	Deep Q Network based Energy Management System
EV	Electric Cehicle
EMS	Energy Management System
FQI	Fitted Q Iteration
EU	European Union
K-BTMKR	Kernel based Boosting based Transfer Multiple Kernel Regression
M-BTMKR	Model based Boosting based Transfer Multiple Kernel Regression
LP	Linear Programming
LSTM	Long Short-Term Memory
MKL	Multiple Kernel Learning
MKBoost	Multiple Kernel Boosting
NFQ	Neural Fitted Q Iteration
NFQEMS	Neural Fitted Q based Energy Management System
PV	Photovoltaic
PT	Total power consumption
RL	Reinforcement Learning
RLEnergy	Reinforcement Learning Energy Simulator
RNN	Recurrent Neural Network
RTP	Real Time Price
SAA	Sample Average Approximation
SAB	Stand-Alone Battery
SP	Stochastic Programming
SPLET	Stochastic Programming and Load forecasting for Energy management with Two stages
SOC	State of Charge
SVR	Support Vector Regression
TOU	Time-of-Use

Acronyms	**Descriptions**
US	United States
USD	US Dollar
V2B	Vehicle-to-Building
V2G	Vehicle-to-Grid

Nomenclature

Nomenclature for Chapter 2

Indices:

i	Electric vehicle
t	Time slot
o	Intraday scenario

Parameters:

K	Total number of scenarios
L	Duration for every time slot (h)
M	Total number of employee EVs
N	Total number of time slots
$\mathcal{SI}_i^o$	Initial SOC (State of charge) for EV i under scenario o (%)
u_i^o	Energy demand for EV i under scenario o (kWh)
$\mathcal{A}_i^o$	Arrival time for EV i under scenario o (h)
$\mathcal{D}_i^o$	Departure time for EV i under scenario o (h)
$\mathcal{M}_{i,t}^o$	Timing constraints for EV i at time slot t under scenario o
$\mathcal{P}_t^b$	Base load power consumption at time slot t (kW)
$\hat{\mathcal{P}}_t^b$	Predicted base load power consumption at time slot t (kW)
$\mathcal{P}_t^v$	PV system output at time slot t (kW)
$\hat{\mathcal{P}}_t^v$	Predicted PV system output at time slot t (kW)
$\mathcal{P}_t^{pc}$	EV charging consumption for public EVs at time slot t (kW)
$\mathcal{SI}_{i,t}$	Initial EV SOC for EV i with revealed information until time t (%)
$\mathcal{A}_{i,t}$	Arrival time for EV i with revealed information until time t (h)
$\mathcal{D}_{i,t}$	Departure time for EV i with revealed information until time t (h)
$S_i^{\min}$	Minimal allowed SOC for EV i (%)
$S_i^{\max}$	Maximal allowed SOC for EV i (%)
$S^{\min\mathrm{A}}$	The lower bound for employee EV SOC for advance departure
I_t^a	Electricity price for time slot t in day-ahead market ($/kW)
I_t^p	Electricity cancelation penalty for time slot t ($/kW)
I_t^r	Electricity price for time slot t for time-of-use electricity ($/kW)
$P_i^{c,\max}$	The maximum charging and discharging rate for EV i (kW)
$P^{ba,\max}$	The maximum charging and discharging rate for stand-alone battery system (kW)

$P_t^{\max}$	The upper limit of the total power consumption (kW)
E_i	Battery capacity for EV i (kWh)
C_i^r	Battery replacement cost for EV i (\$)
G_i	Battery degradation factor for EV i (%)
B	Stand-alone battery system capacity (kWh)
LF	Stand-alone battery system lifetime (h)
η_e	EV battery conversion efficiency (%)
η_b	Stand-alone battery system conversion efficiency (%)

Variables:

p_t^a	Purchased power from day-ahead market at time t (kW)
p_t^u	Power from day-ahead market at time t (day-ahead under scenario o) (kW)
$p_t^{u,o}$	Used power from day-ahead market at time t under scenario o (kW)
p_t^r	Actual used power from time-of-use electricity at time t (kW)
$p_t^{r,o}$	Power from real-time market at time t under scenario o (kW)
$p_{i,t}^c$	Charging rate for EV i at time t (kW)
$p_{i,t}^{c,o}$	Charging rate for EV i at time t under scenario o (kW)
$p_{i,t}^d$	Discharging rate EV i at time t (kW)
$p_{i,t}^{d,o}$	Discharging rate EV i at time t under scenario o (kW)
$p_t^{ba,c}$	Stand-alone battery charging rate at time t (kW)
$p_t^{ba,c,o}$	Stand-alone battery charging rate at time t under scenario o) (kW)
$p_t^{ba,d}$	Stand-alone battery discharging rate at time t (kW)
$p_t^{ba,d,o}$	Stand-alone battery discharging rate at time t under scenario o (kW)
$s_{i,t}$	SOC for EV i at time t (%)
$s_{i,t}^o$	SOC for EV i at time t under scenario o (%)
c_i^e	Ageing cost for EV i (\$/kW)
$c_i^{e,o}$	Ageing cost for EV i under scenario o (\$/kW)
c^{ba}	Ageing cost for stand-alone battery system (\$)
$c^{ba,o}$	Ageing cost for stand-alone battery system under scenario o (\$)
$\mathbf{p^a}$	Power purchase from day-ahead market (kW)
$\mathbf{p^b}$	Base load power consumption vector (kW)
$\mathbf{p^c}$	Employee EVs charging power consumption (kW)
$\mathbf{p^{pc}}$	EVs charging power consumption vector (kW)
$\mathbf{p^r}$	Power bought with time-of-use price (kW)

Nomenclature for Chapter 3

Indices:

m	Kernel
n	Data set
t	Training iteration
s	Source domain

Parameters:

M	Total number of kernels
N	Total number of data set
S	Total number of source domains
T	Total number of training iterations

Variables:

a_n	Coefficient for each sample
$a_{t,n}$	Coefficient for each sample at iteration t
e_m^t	Fitted error for m-th kernel at t-th iteration
$e_{s,m}^t$	Fitted error for m-th kernel from s-th domain at t-th iteration
$\mathcal{H}$	Learned kernel functions set
$\mathcal{H}_K$	Reproducing kernel Hibert space induced by kernel K
k_m	m-th kernel
$\mathcal{K}$	Kernel functions set with size of M
$\mathcal{L}$	Loss function
η_m	Coefficient for m-th kernel
$N\prime$	Size of sub sample of training set
η_m^s	Coefficient for m-th kernel with s-th source domain
ρ^t	Step size for t iteration
ϵ	A small step size
r_n^t	Negative gradient of F at t-th iteration with n-th training data
SN	Total number of data set in all source domains
$\beta_{\mathcal{S}}$	Coefficient vectors for target domain
$\beta_{\mathcal{T}}$	Coefficient vectors for target domain
x_n	Features for training data with index n
$\hat{y}_i$	Predicted value for i-th data set
y_n	Labels (Power consumption) for training data with index n

Nomenclature for Chapter 4

Indices:

t	time step index
j	Auxiliary index for time step
h	Training iteration
k	Training episode

Parameters:

H	Total number of iterations
K	Total number of episodes
T	Total time of time steps
$B_{ev,cap}$	EV battery capacity
$B_{hb,cap}$	Home battery capacity
η_e	EV battery conversion efficiency (%)
η_b	Stand-alone battery system conversion efficiency (%)
γ	discount factor

Variables:

a_t	Action taken at time step t
s_t	State at time step t
r_t	Immediate reward at time step t
A	a finite set of actions
S	a finite set of states
T	Transition function
R	Reward function ($)
G_t	Expected accumulative return at time step t
$H_{soc,t}$	State of charge for home battery at time step t
$P_{v,t}$	Renewable energy generation at time step t (kW)
P_t	Electricity price at time step t ($/kW)
$P_{base,t}$	Base load power consumption at time step t (kW)
$E_{a,t}$	EV charging availability at time step t
$E_{ttd,t}$	The number of hours left before the EV departure
$E_{soc,t}$	EV battery state of charge at time step t
$C_{ev,t}$	Charging or discharging power for EV battery at time step t (kW)
$C_{hb,t}$	Charging or discharging power for home battery at time step t (kW)
$U_{buy,t}$	Energy we need to buy from the grid at time step t (kW)

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Most of the energy consumed today is provided by fossil fuels (including oil, natural gas, and coal). Fossil fuel based energy accounted for 85.0% of the world energy consumption in 2016 [1], and 81.59% of the United States energy consumption in 2016 [2]. The sustained consumption of coal, natural gas, petroleum and other non-renewable energies has caused several serious environmental issues such as greenhouse gas emission (e.g., CO_2) which has led to global warming problems and severe impact on ecosystems and society. Greenhouse gas emission have recently reached its highest level in history [3]. Residential consumption and transportation are two main sectors for energy usage and greenhouse gas emission. As shown in Figure 1.1, nearly 22% of the total energy of United States is consumed by the residential sector and 24% of the total energy is consumed by the transportation sector [4].

To tackle these challenges, many countries are trying to increase renewable energy adoption, promote transportation electrification, and improve the smart grid energy efficiency. European Union has set targets to reduce 40% of the total greenhouse gas emission, improve the usage efficiency of energy by 27%, and realize 27% of all energy usage from renewable energy by 2030 [5]. Besides, it is estimated that 23% of all vehicles will have an alternating powertrain in 2020 and electric vehicles (EVs) are on track to accelerate to 54% of new car sales by 2040 [6]. As shown in Figure 1.2, EV penetration has already been very high for some big cities in 2016 [6]. By integrating modern technologies for communication tools, computation tools and electronic devices, smart grids can now enable better energy transmission and usage efficiency.

However, the increasing adoption of renewable energy and transportation electrification

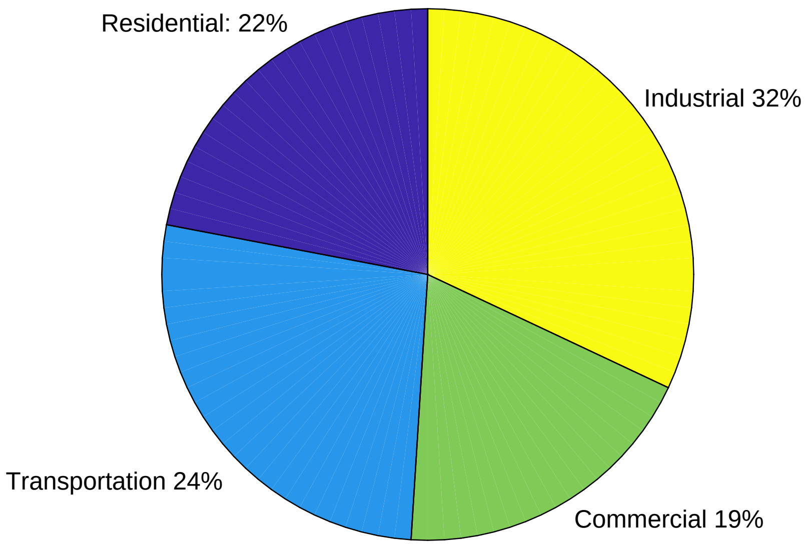

Fig. 1.1 Energy usage for different sectors

has also caused several challenges for power systems. Firstly, the output of renewable energy has a stochastic nature which makes it very difficult to predict and manage. Secondly, the increasing adoption of EVs will bring in a significant power demand on the power system which may damage the grid infrastructure. Thirdly, the adoption of EVs may lead to power flow and quality issues for the grid network [7]. This effect could be even worse for some residential network, where early EV adopters may exhibit similar consumer behaviors in the communities and appear in clusters [8].

The power grid can be divided as supply-side and demand-side. On the supply-side, power starts from the power plant, goes through a transmission network, and is fed hundreds of substations. Demand-side mainly refers as to end-use consumers, starting from substations down to distribution networks. In this thesis, we are trying to tackle the challenges of energy management and short-term electricity load forecasting for the demand-side of the smart grid.

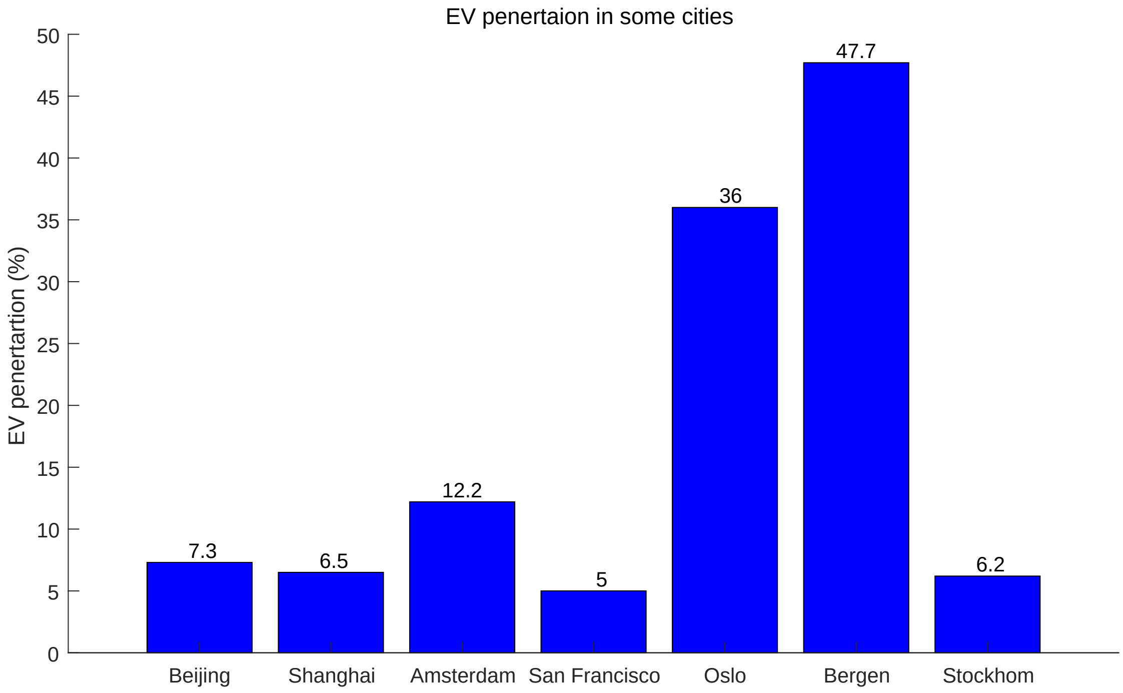

Fig. 1.2 Electric vehicle penetration in some cities

1.1 Smart Grid Concept

The power grid is a distributed complex electrical network which can deliver energy from power plant to different end-use consumers, e.g., residential consumers, industrial consumers, commercial consumers, and electric vehicle users. The power grid has different kinds of components including generators, transformers, transmission lines, substations, and feed lines. According to the raw energy sources used by the power plants, there are different kinds of power plants. Currently, natural gas, coal, uranium and renewable energy are four main raw materials used for power plants in the United States in 2016 [2] as shown in Figure. 1.3. We can see that more than 60% of the electrical energy is generated by fossil fuel sources.

Electric vehicles are more efficient than traditional Internal Combustion Engine (ICE) vehicles [9] and operation of EVs generate less pollutants than ICE vehicles. However, when the whole generation process is considered, electrical energy also generates a large amount of pollutant, e.g, greenhouse gas emission. Thus, it is of significant importance to bring

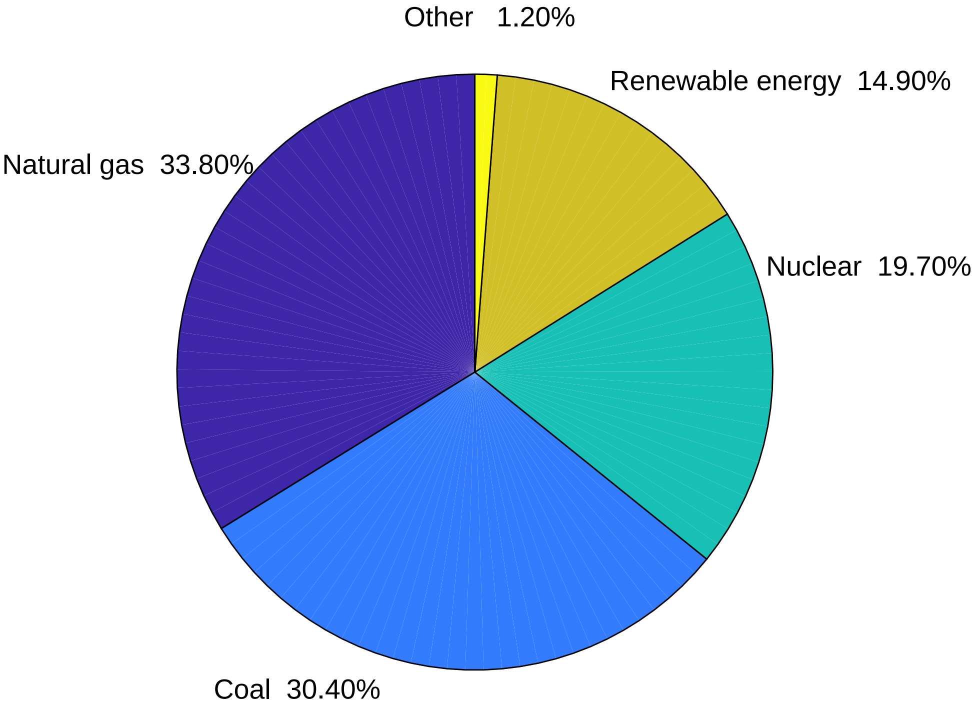

Fig. 1.3 Power plant energy sources in US in 2016

technology innovations into the power grid and improve the energy efficiency of different sub-systems of the power grid.

The smart grid is a complex system that incorporates modern information, control, communication, metering technologies and allows bi-directional communication between utility companies and end-use consumers. As shown in Figure. 1.4, smart grid is an integration of several electrical, communication, computation, and intelligent components. The objective is to make both the energy generation, and energy usage side to be efficient, effective and environment friendly.

The main targets for the smart grid are to reduce the energy waste and improve the grid stability. The bidirectional communication, advanced meters, distributed generation and energy storage can help the smart grid reach such targets. With bidirectional communication, smart grid can have a better observation for the whole network and make better

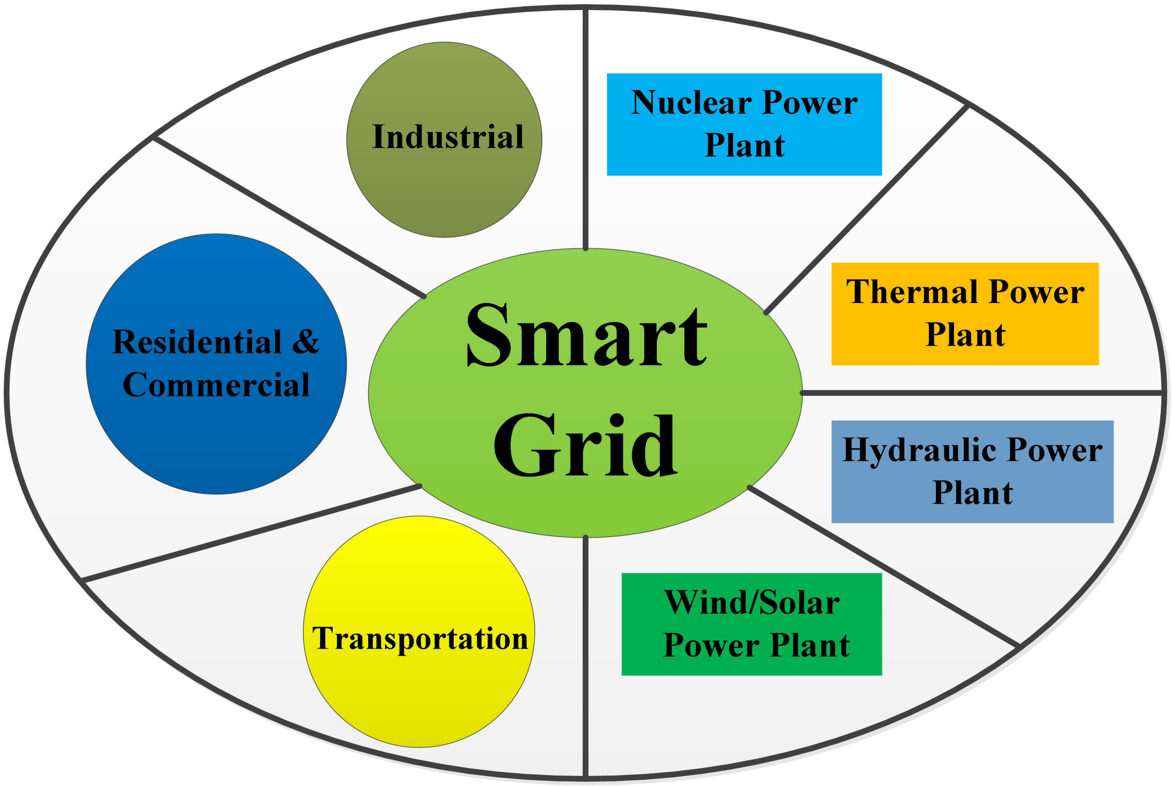

Fig. 1.4 Smart grid components

control decisions. Battery storage can help keep power balance and allow better flexibilities for the smart grid. With the widespread of Electric Vehicles, EV batteries can also be considered as distributed storage which can further help improve the grid stability.

Recent works [10, 11, 12] have shown that machine learning is now playing a more and more important role for the smart grid. With machine learning algorithms, the future electricity demand can be forecasted which will enable a better power generation organization. It is also shown in [10, 11] that reinforcement learning based control algorithms can enable energy management control without requirement for specific knowledge on system dynamics. Machine learning algorithms have also shown a better fault detection performance for the grid networks [12].

1.2 Demand Response

Most of the advantages of smart grid are, in fact, due to its capability of bi-communication, real-time control, and consumers' active responsiveness [13]. Accordingly, demand side management(DSM), including everything that is done on the demand side, is an important and integral part of the smart grid. According to the definition given by the the US Department of Energy (DOE) [14], demand response means changes in electricity usage by end-use consumers from their normal consumption patterns in response to changes in the price of electricity over time, or to incentive payments designed to induce lower electricity use at time of high wholesale market prices or when system reliability is jeopardized. The main participants for demand response include utility companies, end-use consumers, and aggregator. There are two types of demand response: price-based demand response and incentive-based demand response. Price-based demand response is a kind of indirect control, in which the price signal serves to incentivise end-use consumers. For incentive-based demand response, end-users will receive direct payment from utility companies when participating in a particular demand response program. The main reasons for utility companies to support demand response programs are that the peak power generation is very expensive and high peak power consumption may cause potential stability issues for power networks.

Demand response programs were previously mainly focused on large end-use consumers, e.g., large commercial and industrial consumers. These consumers usually have necessary components including battery storage, advanced meters and energy management systems. With the development of communication, computation and energy storage technologies, more and more individual consumers are now beginning to participate in the demand response programs. Flexible loads are the critical components for demand response and the main focus for demand response is trying to harness the flexibilities of these loads while controlling the negative impacts for end-use consumers. There are mainly three different types of flexible loads including: controllable home devices, stand-alone batteries, and thermostatic load. EV batteries can also be treated as battery storage and used to participate in demand response programs. In this thesis, our work is mainly focused on the demand-side of smart grid.

1.3 Machine Learning and Applications for Smart Grid

There are several definitions for machine learning [15, 16, 17]. As defined in [17], machine learning is the field that is concerned with the question of how to construct computer programs that can automatically improve with experience. In general, there are three main categories for machine learning: supervised learning, semi-supervised learning and unsupervised learning.

In this thesis, we explore how smart grid energy management and electricity load forecasting can benefit from machine learning algorithms. Supervised learning, reinforcement learning, and transfer learning are used in our three pieces of work. Hence, we introduce the basic concepts for three main machine learning categories and transfer learning as follows.

Supervised Learning algorithms try to learn a mapping function which maps input to output based on a given dataset. The given dataset consists of a set of labeled data. With the learned function, the system can make predictions for the output values with any new given inputs. Classical problems for supervised learning are: classification and regression.

Unsupervised Learning algorithms are used when we try to infer a function to describe a hidden structure from a set of unlabeled data. The center problem for unsupervised learning is clustering.

Reinforcement Learning (RL) setting is quite different from supervised learning and unsupervised learning. The fundamental idea of reinforcement learning is learning by interactions. A RL agent interacts with the environment at discrete time steps. The goal for RL algorithms is to learn a policy that could maximize the long term expected return for the RL agent.

Transfer Learning algorithms try to apply the knowledge learned from one or more tasks to other related task to accelerate the learning process or improve the learning performance. Both the learned models or training instances could be transferred. When only a limited amount of data is available for a domain we are interested (target domain) and a large amount of data is available for other related domains (source domains), it will be desirable if some knowledge learned from source domains could be the transferred to the target domain.

There are many practical applications for machine learning. Recent work of applying machine learning on smart grid [10, 11, 12, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27] have shown the effectiveness of machine learning for the smart grid. With machine learning algorithms, the forecasting accuracy for future electricity power consumption can be improved which will enable a better power plant generation organization. It is also shown in [10, 11, 21] that reinforcement learning based control algorithms can enable energy management control without requirement for specific knowledge on system dynamics. Machine learning algorithms have also shown a better fault detection performance for the grid networks [12, 22, 23].

In [24, 25], the authors demonstrate that power quality could be improved with machine learning techniques. It is also shown in [26, 27] that machine learning could assist the design of electricity price with forecasting of consumers' behaviors and power consumption.

1.4 Research Objectives and Methodologies

Smart grid development has drawn a lot of attention since it will enable efficient and environmentally friendly energy usage. However, with the increasing adoption of renewable energy and EVs, more challenges have been brought to the smart grid. Building sector accounts for 40% of total US energy generation in 2016 [28]. Thus, it is of significant importance to improve the energy efficiency of buildings. However, there are several uncertainties when designing an energy management control strategy. With the development of home based energy storage and controllers, energy management for residential sectors has attracted more and more attention [29]. But we may not have accurate system dynamics for the residential house. Meanwhile, for some newly built houses or buildings, we may not have enough data which makes it hard to train a reliable load forecasting model.

In this thesis, we are trying to tackle the challenges for energy management systems and short-term electricity load forecasting for office buildings and residential houses. Mainly, there are four types of methods to deal with energy management for the smart grid: rule-based methods, model-based methods, model-free methods, and model-assisted methods. Rule-based methods entailed by a set of heuristic strategies are based on engineering knowledge of the system. Model-based methods require explicit knowledge of system dynamics to find a control strategy. Model-free methods do not require any knowledge on the system dynamics. Model-assisted method is a hybrid method of model-based and model-free method.

Different methods could work for electricity load forecasting including statistical meth-

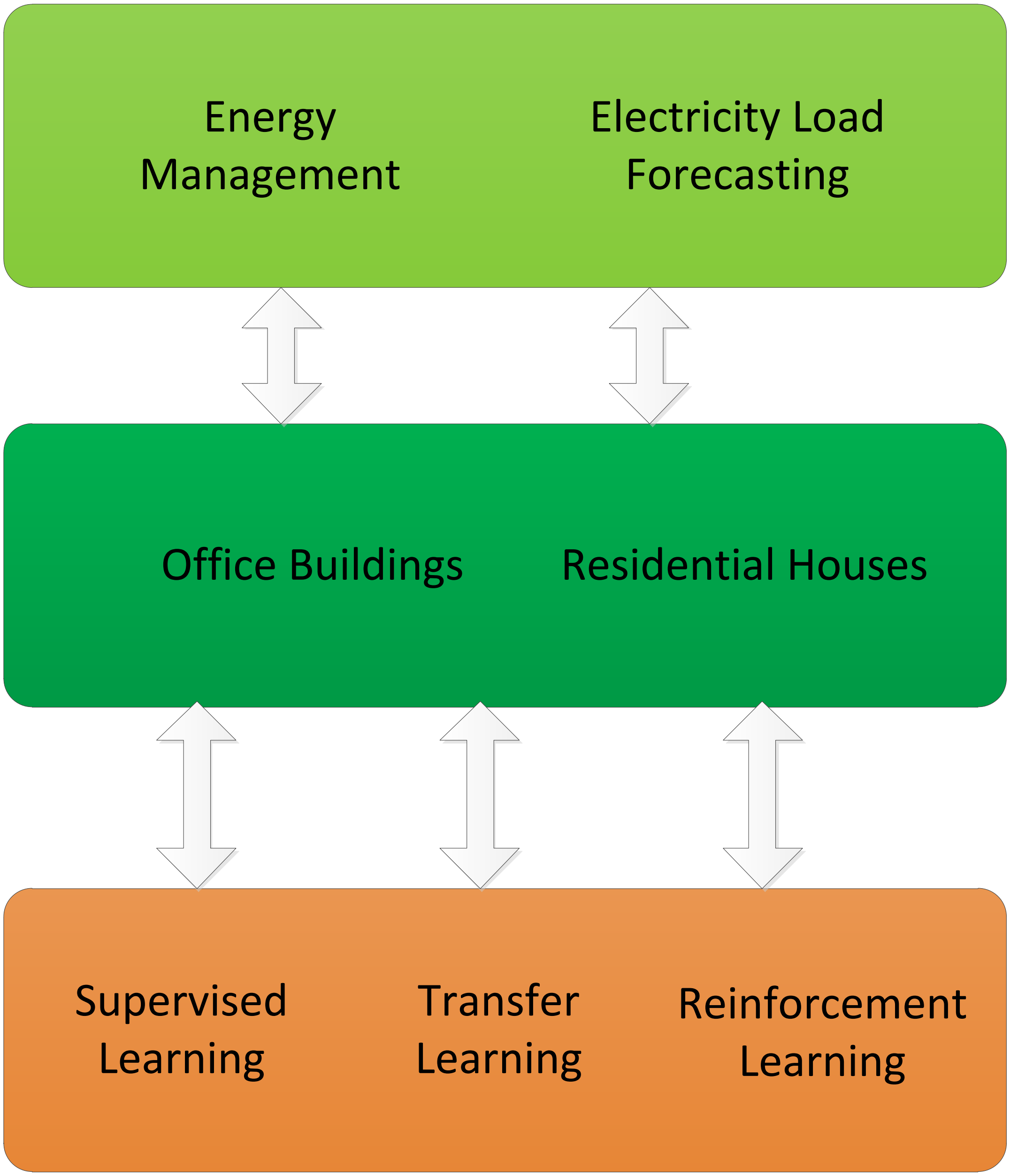

Fig. 1.5 Objectives and machine learning algorithms.

ods, time series analysis and machine learning algorithms. The overall objective of this research is to investigate how can a sustainable smart grid benefit from modern machine learning algorithms. As shown in Figure 1.5, supervised learning, transfer learning and reinforcement learning are used in this thesis to tackle the challenges for energy management and electricity load forecasting for office buildings and residential houses. More specific objectives and methodologies discussed in this thesis are summarized below.

- In the first work, we deal with the day-ahead and intra-day energy scheduling for office buildings with the emerging trend of transportation electrification and renewable energy generation. Renewable energy generation and vehicle driving patterns are inherently stochastic. In this work, we tackle the energy management for office building with a two-stage stochastic programming based framework with short-term electricity load forecasting.

- In the second work, we investigate how to improve the performance for short-term electricity load forecasting especially when only limited data are available. We use machine learning, specifically multiple kernel learning methods for electricity load forecasting. We further use transfer learning to reuse the knowledge learned from other sources to improve the forecasting performance when only limited data is available for the target domain we are interested in.

- In the third work, we study the benefit of model-free reinforcement learning for home energy management when only little knowledge of system dynamics is available. We use both off-line and on-line reinforcement learning algorithms to tackle home energy management problems.

1.5 Thesis Organization and Contributions

In this thesis, we present three pieces of work to exemplify our approaches for energy management and electricity load forecasting. The current thesis is structured as follows:

In Chapter 2, we deal with the building energy management problem and present a stochastic programming based framework integrated with predicted electricity demand for large office building. Contributions for this chapter are summarized as follows:

- A new energy scheduling scheme is proposed to coordinate the integration of large office buildings with PV panels, a large number of EVs, day-ahead power market and time-of-use electricity to optimize the energy management for the building.

- Two computationally efficient control algorithms, Stochastic Programming and Load forecasting for Energy management with Two stages (SPLET) and Sample Average

Approximation based SPLET (SAA_SPLET) are proposed which can help reduce the operation cost for the office building. The uncertainties lie in both the demand-side and supply-side: the base load power consumption, renewable energy output, EV arrival time, EV departure time, and EV arrival state of charge (SOC) are addressed in the proposed algorithms. The effectiveness of the proposed algorithms are presented with real-world data.

- EV Vehicle-to-Building (V2B) and stand-alone battery systems are considered as countermeasures for the mismatch between day-ahead power scheduling and actual real-time demand.

The work described in Chapter 3 is published as: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, Cham, 2017: 39-51. And a journal version of this paper is in preparation to be submitted to IEEE Transactions on Smart Grid.

In Chapter 3, we look into the single home short-term electricity load forecasting problem. We use gradient boosting framework to improve the electricity load forecasting accuracy and use transfer learning to deal with new homes or buildings with limited available data. In this chapter, we have the following contributions:

- We propose a boosting-based learning framework to learn regression models with multiple kernels efficiently.
- We leverage the Multiple kernel learning (MKL) models learned from other domains with transfer learning to deal with electricity demand forecasting when only limited data is available.
- Negative transfer between target domain and source domains is analyzed which shows that the proposed algorithm can prevent potential negative transfer.

In Chapter 4, we investigate how to use model-free reinforcement learning algorithms for home energy management with potentially unknown system dynamics. Contributions for this chapter are summarized below:

- We propose to formulate the home energy management including EV charging as a Markov Decision Process (MDP).

- We tackle it with two model-free reinforcement learning algorithms: Neural Fitted Q Iteration (NFQ) and Deep Q-Network (DQN), and we investigate their performances on reducing operating cost and peak power consumption with real-world data.

In Chapter 5, we conclude on the problems and approaches investigated in this thesis and discuss some potential research directions for future work.

1.6 Contributions of Collaborators and Publications

The contributions mentioned above are included in the following published or to be submitted peer reviewed papers. The research presented in this thesis also benefits from the collaborations with other scholars especially from the co-authors of each paper. The notable contributions from each collaborators are summarized for each paper that contributes to this thesis. The candidate (Di Wu), had done the core for the research work demonstrated in this thesis, including: collection and pre-processing of the data, design of the algorithms, implementation of the evaluation, analysis of the results, and the writing of the paper, except for the work done by collaborators described bellow.

- Di Wu, Haibo Zeng, Chao Lu, Benoit Boulet, Two-Stage Energy Management for Office Buildings with Workplace EV Charging and Renewable Energy, IEEE Transactions on Transportation Electrification, 2017, 225-237.
    - Prof. Chao Lu: Gave some suggestions for problems definition.
    - Prof. Haibo Zeng and Benoit Boulet: Supervision for the whole work and editing of the manuscript.
- Di Wu, Boyu Wang, Doina Precup, Benoit Boulet, Boosting Based Multiple Kernel Learning and Transfer Regression for Electricity Load Forecasting, Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, Cham, 2017: 39-51
    - Boyu Wang: Offered constructive suggestions for transfer learning framework design and helped edit the manuscript.
    - Prof. Doina Precup and Benoit Boulet: Supervision for the whole work and editing of the manuscript.

- Di Wu, Boyu Wang, Doina Precup, Benoit Boulet, Gradient Boosting Based Transfer Regression for Smart Grid Demand Prediction, in preparation for submission to IEEE Transactions on Smart Grid.
  - Boyu Wang: Offered constructive suggestions for negative transfer analysis.
  - Prof. Doina Precup and Benoit Boulet: Supervision for the whole work and editing of the manuscript.
- Di Wu, Guillaume Rabusseau, Vincent François-Lavet, Doina Precup, Benoit Boulet, Optimizing Home Energy Management and Electric Vehicle Charging with Reinforcement Learning is accepted by Adaptive Learning Agents (ALA) workshop, 2018
  - Guillaume Rabusseau: Offered constructive suggestions for state space analysis of MDP.
  - Vincent François-Lavet: Offered constructive suggestions for design the control strategy for home energy management.
  - Prof. Doina Precup and Benoit Boulet: Supervision for the whole work and editing of the manuscript.

During the Ph.D., some related work has also been conducted, these work can be treated as complementary work for this thesis, and the publications are listed as follows:

- Di Wu, Haibo Zeng, Benoit Boulet. Impact Analysis of EV Charging with Mixed Control Strategy, Journal of Energy and Power Engineering, 2015: 731-740.
- Di Wu, Haibo Zeng, and Benoit Boulet. Impact Analysis of Controllable Home Appliances and EVs on Neighborhood Level Network with Smart Control, Electric Vehicle Symposium and Exhibition (EVS 29), 2016: 543:552.
- Di Wu, Haibo Zeng, and Benoit Boulet. Neighborhood Level Network Aware Electric Vehicle Charging Management with Mixed Control Strategy[C], Electric Vehicle Conference (IEVC), 2014 IEEE International. IEEE, 2014: 1-7.
- Cheng Wang, Di Wu, and Haibo Zeng. Wang C, Wu D, Zeng H, et al. CPS based Electric Vehicle Charging Directing System Design in Smart Grid[C], Innovative Smart Grid Technologies Conference (ISGT), 2016 IEEE Power & Energy Society. IEEE, 2016: 1-5.

- Xiong Jian, Di Wu, Haibo Zeng and Xiaoyu Wang. Impact Assessment of Electric Vehicle Charging on Hydro Ottawa Distribution Networks at Neighborhood Levels[C], Electrical and Computer Engineering (CCECE), 2015 IEEE 28th Canadian Conference on. IEEE, 2015: 1072-1077.

Chapter 2

Two-Stage Energy Management for Office Buildings with Workplace EV Charging and Renewable Energy

2.1 Introduction

With the progressive exhaustion of fossil energy and increasing concerns on protecting the environment, electric vehicles (EVs) are considered as a more sustainable solution to serve people's mobility needs, and hence their adoption growing very fast. Many countries are working to promote the development of transportation electrification. It is expected by the International Energy Agency that the annual growth rate for EV market will be higher than 20% before 2020 and the electric vehicle stock will reach 20 millions in 2020 and 140 millions in 2030 [30]. The benefits of EVs toward zero emission mobility can be maximized when they are charged with renewable energy sources such as solar, wind, and hydroelectricity [31]. This vision could be partially realized in buildings, as more and more buildings are integrated with solar photovoltaic (PV) panels to provide part of daily power consumption.

Workplace EV charging service is considered of crucial importance to promote EV adoption by company employees and will be beneficial for employers, employees, and communities [32]. It is shown in a home parking garage survey that only 40% of people in the US have such garages in which home chargers could be installed [33]. Right now, more and more companies are starting to provide workplace charging service. It is shown that when workplace charging service is provided, the employees will be 20 times more likely to drive an EV [34].

In this chapter, we address the problem of energy scheduling for large office buildings integrated with electric vehicles and photovoltaic systems, and the design of the energy management system (EMS). The EMS takes care of coordination with the power markets and energy scheduling of the building. The office building is assumed to provide charging services for the employee EVs and public EVs. The large power draw of such buildings has sparked interest in providing ancillary services through day-ahead power market and intelligent operation of building EMS [35]. The building operator can get favorable electricity pricing from day-ahead power market. To participate in the day-ahead power market, the building EMS should decide on the energy to purchase for every time slot one day before.

Fig. 2.1 shows the EMS for such a large office building. On the energy supplying side, the building could get energy from the day-ahead power market, time-of-use electricity, and integrated PV systems. On the energy consumption side, the building EMS has to provide energy for electric vehicle charging and other power consumption. In such scenarios, there might be a large number of employee EVs parked in or near the building, and these EVs may arrive and leave at similar times. The challenge for an efficient energy scheduling lies in the uncertainties associated with power generation and consumption, including base load, PV output, EV arrival time, EV departure time, and EV arrival battery state of charge (SOC).

Demand response can help reduce the operating cost and peak power demand [36]. In this chapter, we propose two building energy management control algorithms to deal with uncertainties in power demand: Stochastic Programming (SP) and Load forecasting for Energy management with Two stages (SPLET) and Sample Average Approximation based SPLET (SAA_SPLET) with participation of both day-ahead planning and real-time operation. As the EVs and plug-in hybrid electric vehicles (PHEVs) will be parked for most time of the day [37], the charging power consumption can be treated as flexible load. Both of the two proposed control algorithms have two stages. In the first stage (i.e., day-ahead scheduling), we use prediction techniques for base load power consumption, PV output, and take advantage of known probability distributions of employee EV arrival time, departure time, and arrival SOC. In the second stage (i.e., real-time operation), with the given power purchased from day-ahead power market, the EV charging is scheduled to reduce the total

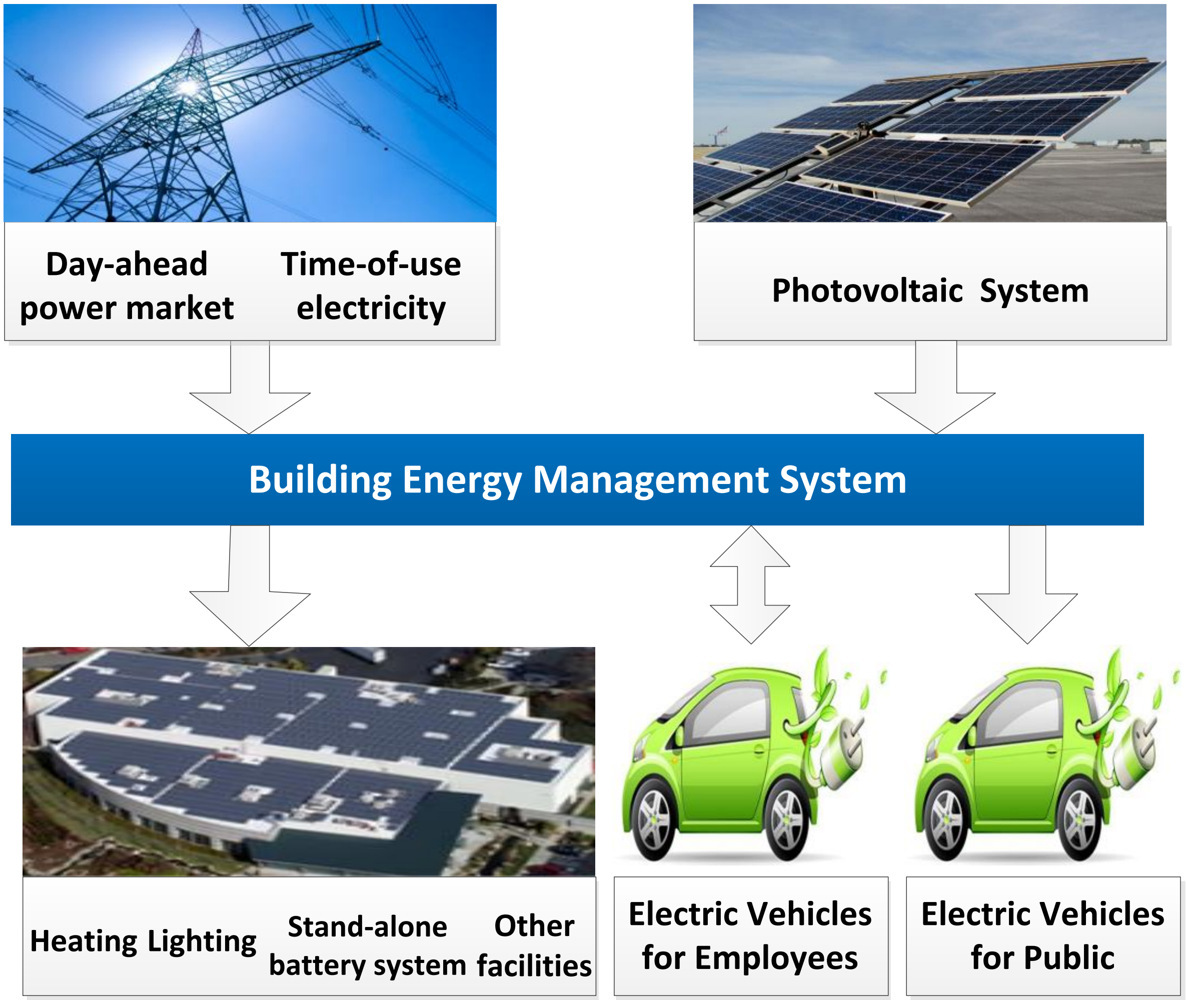
Day-ahead power market
Time-of-use electricity
Photovoltaic System
Building Energy Management System
Heating
Lighting
Stand-alone battery system
Other facilities
Electric Vehicles for Employees
Electric Vehicles for Public

Fig. 2.1 Energy management system for a large office building.

operating cost with new information (e.g., EV arrival, their arrival battery SOC) for every time slot. In addition, we use vehicle-to-building (V2B) and stand-alone battery (SAB) system as countermeasures for the mismatch between the day-ahead power scheduling and real-time power demand.

2.2 Related Work

Energy management systems play a key role for the energy efficiency of buildings and homes [38, 39]. Several recent studies are conducted on the energy management of smart building integrated with EVs and/or PV system. In [40], an EV charging control algorithm for buildings integrated with PV system is presented. It proposes to schedule the EV charging based on the predicted base load power consumption and PV system output. But this paper does not consider the uncertainties for EV charging demand. In [41], a day-ahead energy scheduling algorithm for smart building integrated with distributed renewable generation is proposed, which can efficiently reduce the greenhouse gas emissions. However, this paper does not consider the flexibility of EV charging demand and does not discuss the operating cost for the building. Different from the above works, this chapter proposes two energy management control algorithms for buildings integrated with PV system, EVs, day-ahead power market and time-of-use electricity.

Transportation electrification will increase the power consumption of buildings. The impacts for charging demand of EVs and PHEVs on power systems are investigated in a number of papers [9, 42, 43], In [9], an overview for the impacts of EVs on distribution network is presented. In [42], the authors use a probabilistic method to study the impacts of EV charging on the power system, using metrics including network demands, network voltage levels, and secondary transformer overloading. The effects of large-scale utilization of EVs on the total system loss are studied in [43]. The prediction of EV charging demand is discussed in [44, 45, 46]. In [44], an EV parking lot allocation method for charging is presented considering the EV charging demand and driving behavior. In [45], the authors present how to use autoregressive integrated moving average method to forecast both the conventional electrical load and EV charging demand. In [46], a general test case for the transportation electrification research is presented.

Control strategies for electric vehicle charging have been discussed in several recent papers [47, 48, 49, 50, 51]. In [47], the authors presented a survey on EV charging. In [48], a control algorithm for EV charging and its hardware implementation is discussed. In [49], a multi-stage control scheme for islanded microgrid with EVs is discussed. In [50], the authors present an optimal scheduling method for microgrid with EV fleets and data center storage systems. In [51], a vehicle-originating-signals approach is proposed for the charging control of EVs which can help reduce the communication overhead with minor effect on performance. In[52], a hierarchical model predictive control based energy management system is proposed for smart grid with integration of EVs. In [53], an EV charging control scheme is proposed for EV charging station that considers the output of renewable energy and energy storage system. In [54], the authors present a study on different control priority criteria on the chargeability of EVs and charging fairness.

Meanwhile, EVs can be treated as distributed storage and help improve the grid load profiles [55, 56, 57, 58]. In [55], the authors show that the electric grid and automobile fleet are surprisingly complementary. This paper also demonstrates that EVs can help stabilize the grid and support the adoption of renewable generation. It is demonstrated in [56] that simultaneous allocation of EV parking lots and distributed renewable resources in distributed grid network could help reduce the system loss. In [57, 58], it is shown that EVs can be used as demand response tool and help support high adoption of renewable energy in the smart grid.

The EV charging demand and output of renewable energy are naturally random. Several recent papers [59, 60] use stochastic programming to deal with these uncertainties. In [59], the uncertainties for the output of wind and solar energy are discussed in a stochastic programming framework. However, this paper does not consider the prediction of base load power consumption and the uncertainties for electric vehicle charging demand. In [60], the authors demonstrate that the uncertainties of EV charging will affect its contribution on system reliability. The stochastic output of renewable energy will also influence the scheduling of EV charging scheduling. In [61], the design of EV charging scheduling is based on the forecasted PV system output. A two-stage framework has been discussed in some previous works [59, 62] for energy management of microgrid. However, the scheduling of flexible power consumption such as EV charging is not well discussed in the first stage of the two-stage framework. In this chapter, we propose two two-stage based control algorithms combining both the stochastic programming and forecasting method for building energy management integrated with large number of EVs and PV panels. The proposed control algorithms try to optimize the building energy management while considering the stochastic nature of both power consumption and power supply.

To participate in the day-ahead power market, the decision on the day-ahead energy transactions has to be made before the actual realization of the power demand, resulting in a typical mismatch between the day-ahead scheduling and actual power demand realization. This kind of mismatch is discussed in several papers, e.g. [59, 63], where both propose to use fast start-up generators to supply complementary energy when such a mismatch occurs. As EVs would be parked in the building for a long period in the day and only need to be charged with enough energy before the departure time, EVs could be treated as flexible energy storage and give energy back to buildings [64]. In this chapter, EV V2B and stand-alone battery system are considered as countermeasures to mitigate the potential mismatch between day-ahead power scheduling and actual demand realization.

2.3 Technique Background for Two-Stage Stochastic Programming

Linear programming (LP) is a fundamental planning tool for deterministic decision making where all necessary information is available. However, for many real-world scenarios, not all the information is available at the time of making decisions. When some decision variables are uncertain, but assumed to lie in a set of possible values, a solution for the objective function over all possible choices of uncertainties is feasible. This is suitable for stochastic programming [65]. Stochastic programming is an approach for modeling optimization problems involving uncertainties. Stochastic programming takes advantage of probability distributions for data that is known or could be estimated. The objective for stochastic programming is to come up with a decision that could be performed well on average. An example is the design of a production plan for consumers with random demand. Here, probability distributions of demand could be estimated from data that have been collected over time. The goal is to find a plan that will result in maximal revenue on average over all possible realizations of random variables.

Two-stage stochastic programming is the most widely studied stochastic programming model. There are two stages for decision making in two-stage stochastic programming framework. In the first stage, one decision is made with some uncertain variables, but assumed to lie in a set of values with a certain probability distribution. In the second stage, another decision will be made upon the decision made in first stage and actual values of random variables. The objective for two-stage stochastic programming is to maximize the average expected return. The optimal policy for two-stage stochastic programming is a single decision for the first stage and a recourse of decisions (decision rules) for second stage responding to every possible actual realization of random variables.

2.4 System Models

In this section, we discuss the model for several components in the proposed framework, including the base load power consumption, electric vehicle charging model and its battery ageing model, stand-alone battery system model, and PV output model. The time in one day is divided into N time slots with equal duration L, such that $NL = 24h$. It is assumed that advanced charging meters are set up for the building, which allow the building EMS to acquire the real-time EV SOC and stand-alone battery system SOC.

2.4.1 Electric Vehicle Model

It is assumed that there are in total M employee EVs for the office building. For employee EV i, its arrival time $\mathcal{A}_i$, initial battery state of charge $\mathcal{SI}_i$, and departure time $\mathcal{D}_i$ are unknown a priori, but their probability distributions can be gathered from historical data or through survey [66]. In this chapter, the charging rate for the employee EVs is assumed to be a real number ranging from zero to its maximum charging rate $P_i^{c,\max}$.

V2B is to discharge the energy stored in EV batteries back to the building which can help reduce the peak power consumption and operating cost. V2B for employee EVs is considered in this chapter, which means that the energy stored in EVs could be discharged back to the building. The charging and discharging of EVs are controlled by building EMS. To evaluate the ageing of EV batteries, we consider the most popular battery technology for passenger cars: the lithium-ion battery as adopted in Tesla Model S and Honda Fit EVs [67, 68]. The degradation cost for charging/discharging is dependent on several factors: the battery replacement cost (C^r), residual battery SOC at the end of life (typically defined as 80%), charging and discharging rate, battery capacity (E_i), and battery degradation factor (G) [69]. Equation (2.1) gives the evaluation of the degradation cost for EVs with lithium-ion batteries [69], where $p_{i,t}^c$ is the EV charging rate for EV i at time slot t and $p_{i,t}^d$ is the EV discharging rate for EV i at time slot t.

$$c_i^e = \frac{C_i^r}{100-80} \cdot \frac{\sum_{t=1}^{N}(p_{i,t}^c + p_{i,t}^d) \cdot L}{E_i} \cdot G_i \tag{2.1}$$

For office buildings, there could be some visitors driving electric vehicles. In this chapter, we assume that the office buildings will provide charging service for the visitor EVs and nearby residential EVs. The above mentioned two kinds of EVs are denoted as public EVs. Different from the employee EVs, the charging demand for public EVs are quite random and these EVs may not be willing to accept charging scheduling. In this chapter, these EVs are assumed to be charged as soon as possible after their arrival (the charging will start at the beginning of the next time slot after arrival).

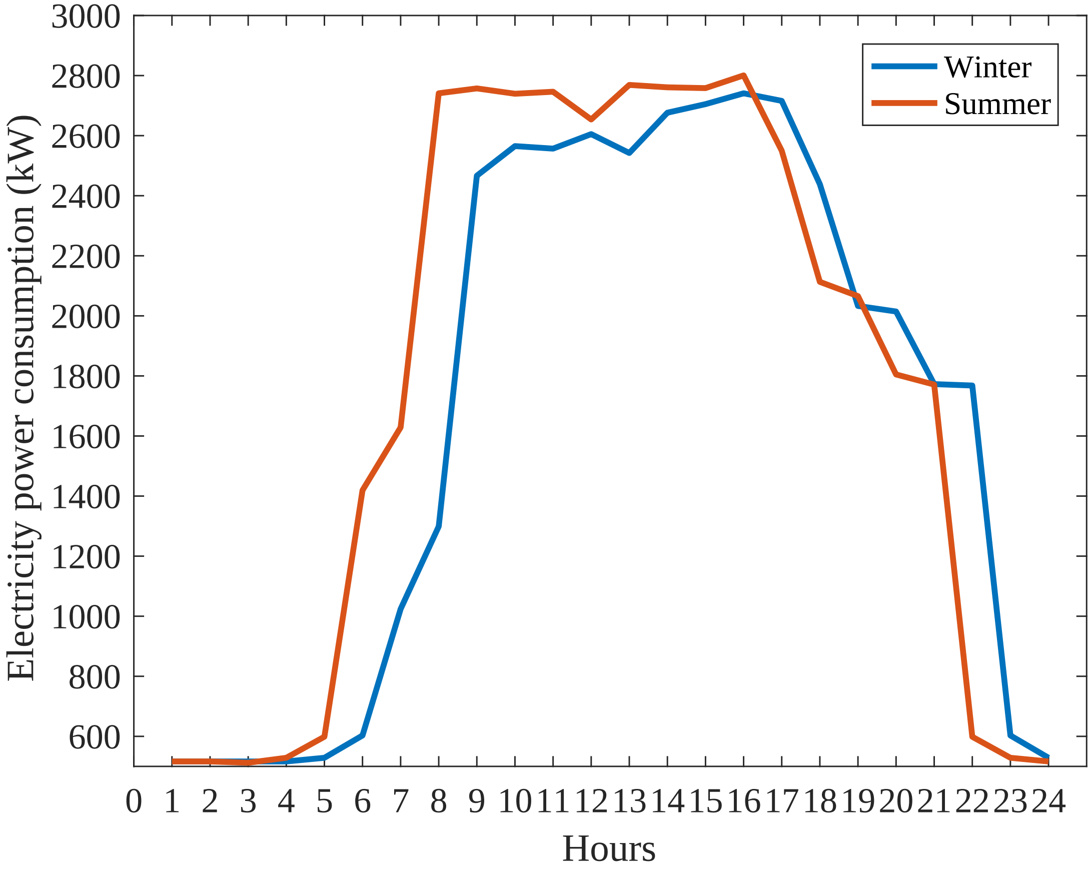

Fig. 2.2 Base load consumption for an office building in two representative days, one in winter, one in summer

2.4.2 Base Load Power Consumption

The power consumption of the office building is divided into three groups: employee EVs charging power consumption $\mathbf{p^c}$, public EVs charging power consumption $\mathbf{p^{pc}}$ and base load power consumption $\mathbf{p^b}$. Base load power consumption includes all other power consumption except the EV charging: facility, cooling, heating, fans, interior lights, interior equipments, etc. Typically, the power consumption for the office building starts to increase at 5am and starts to decrease at 8pm. However, it has been observed that the base load power consumption will fluctuate from day to day. Fig. 2.2 shows the power consumption for a typical winter day (January 1st) and a typical summer day (July 1st) for a large office building in Los Angeles [70].

2.4.3 Stand-alone Battery System Ageing Model

The building can be equipped with a stand-alone battery system, to further improve the building energy efficiency. Lead-acid battery system is a possible option for its low cost in large scale applications. The degradation cost for lead-acid batteries depends on the battery capacity, the battery lifetime (LF), charging rate ($p_t^{ba,c}$), and discharging rate ($p_t^{ba,d}$) [71], as shown in Equation (2.2).

$$c^{ba} = \frac{B}{LF} \sum_{t=1}^{N} (p_t^{ba,c} + p_t^{ba,d}) \cdot L \tag{2.2}$$

2.4.4 Photovoltaic Energy Output Model

PV energy is becoming an important energy resource for large office buildings, as building-integrated photovoltaic panels are increasingly placed on top of the building roof. In Yann's model [72], the power output of a PV system is correlated with the ambient temperature and irradiance level, as shown in Equation (2.3) (in international system of units):

$$\mathcal{P}_t^v = \left(P^{v,\max} \cdot \frac{R_t}{1000} \cdot (1 - \gamma(V_t - 25)) \right) \cdot Z \tag{2.3}$$

where $P^{v,\max}$ is the peak output of the PV module, R_t is the irradiance level at time t, V_t is the cell temperature at time t, γ is the temperature coefficient factor, and Z is the number of PV modules.

In the above equation, the cell temperature V and the irradiance level R are the only dynamic parameters that are dependent on the weather. The PV system output during one hour is assumed to be stable and we use the average output power from the PV modules as the PV output for that time slot.

2.5 Problem and Framework Formulation

To participate in the day-ahead power market, the building EMS should decide the purchase of energy for every time slot one day before. In addition, it needs to respond to real-time power demand (including both base load and EV charging) with real-time grid operation. There are several parameters that are naturally random and unknown a priori, including base load power consumption, PV output, EV arrival time, EV departure time, and initial EV SOC at arrival. This is different from deterministic optimization problems where parameters are known and fixed. Hence, we propose the use of stochastic programming, a framework for modeling optimization problems that involve uncertainty [73]. Stochastic programming takes advantage of the knowledge on the probability distributions of the parameters. Its goal is to find solutions that are feasible for all possible scenarios while minimizing the expected cost [73]. Here a scenario is defined as a possible realization of the random parameters. One potential weakness for stochastic programming framework is the large number of scenarios which demand high computation resources. Here we propose two control algorithms: SPLET and SAA_SPLET, which can significantly reduce the number of scenarios while provide favourable results. The SAA_SPLET is based on SPLET. We will first introduce SPLET and then discuss the differences between the two algorithms in Section 2.5.4.

2.5.1 SPLET: Energy Scheduling with Two Stages

We first give an overview of the proposed SPLET algorithm. As shown in Fig. 2.3, it contains two stages that utilize a stochastic programming framework to deal with uncertainties.

In the first stage, the energy scheduling with the day-ahead power market is solved, which determines, as output, the amount of power to be purchased from the day-ahead market for each time slot in the following day. The input information includes the day-ahead and time-of-use electricity price, historical employee EVs arrival and departure information, historical public EV charging demand and prediction for PV power output, prediction for base load power consumption for all the time slots in the next day.

The second stage is real-time energy scheduling, at the beginning of every time slot t, the EMS will schedule the employee EVs charging and the transactions with time-of-use electricity to minimize the total operating cost. The power purchase $\mathbf{p^a}$ from day-ahead market resulted from the first stage is used as input. It also uses as input the (updated) prediction of the base load, PV output for the remainder of the day. For employee EVs, with the information on arrived/departed EVs and their initial SOC, the scenarios and their probability should be updated. We use $\mathcal{A}_{i,t}$ to denote the arrival time distribution with revealed information until time t for employee EV i. The arrival SOC and departure time can be updated similarly. The optimal employee EV charging scheduling will be determined in this stage. As for the public EVs, the charging demand are quite random and hard to

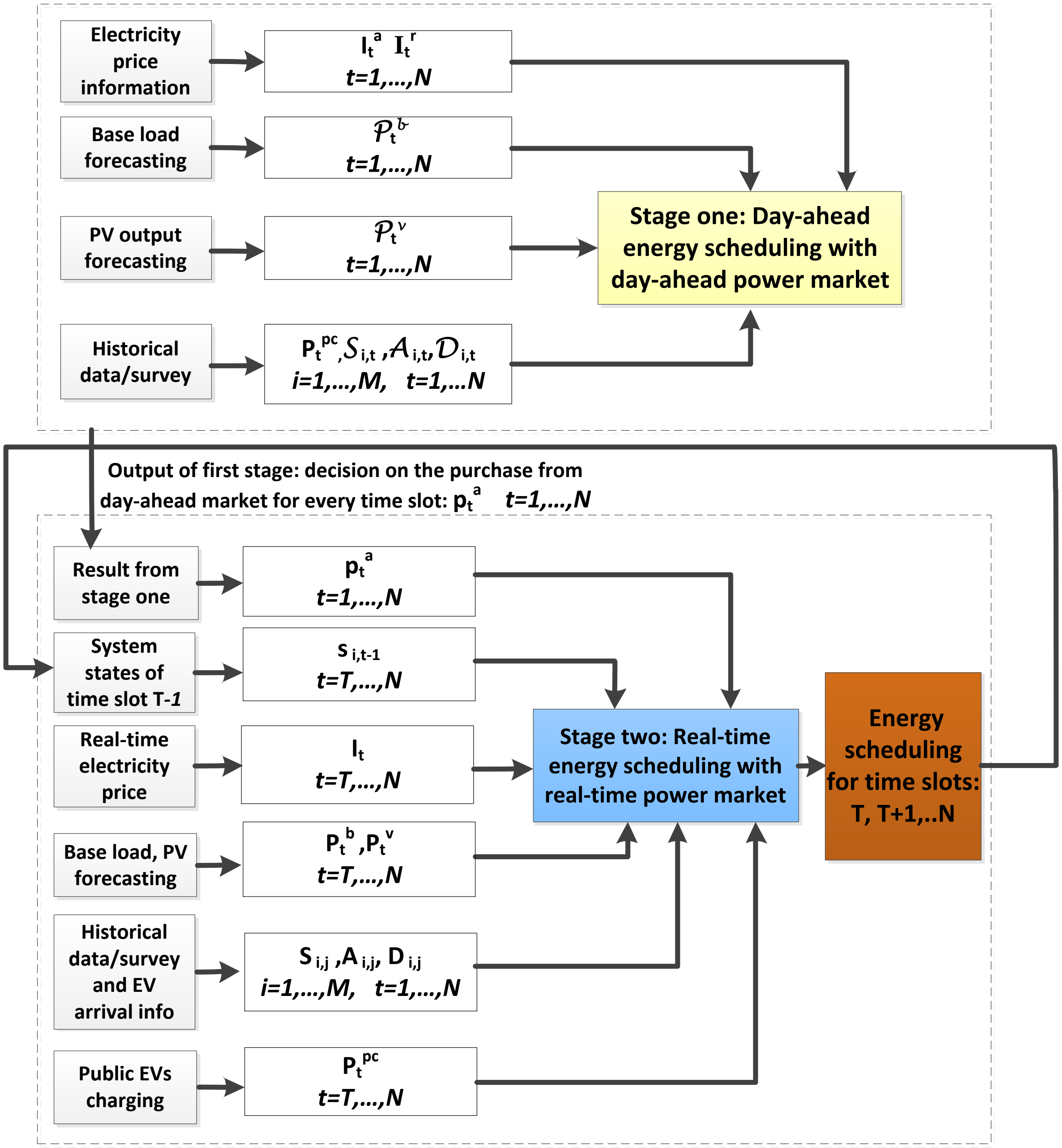
Electricity price information
I_t^a I_t^r
t=1,…,N
Base load forecasting
$\mathcal{P}_t^b$
t=1,…,N
PV output forecasting
$\mathcal{P}_t^v$
t=1,…,N
Stage one: Day-ahead energy scheduling with day-ahead power market
Historical data/survey
$P_t^{pc}, \mathcal{S}_{i,t}, \mathcal{A}_{i,t}, \mathcal{D}_{i,t}$
i=1,…,M, t=1,…N
Output of first stage: decision on the purchase from day-ahead market for every time slot: p_t^a t=1,…,N
Result from stage one
p_t^a
t=1,…,N
System states of time slot T-1
$S_{i,t-1}$
t=T,…,N
Real-time electricity price
I_t
t=T,…,N
Stage two: Real-time energy scheduling with real-time power market
Energy scheduling for time slots: T, T+1,..N
Base load, PV forecasting
P_t^b, P_t^v
t=T,…,N
Historical data/survey and EV arrival info
$S_{i,j}, A_{i,j}, D_{i,j}$
i=1,…,M, t=1,…,N
Public EVs charging
P_t^{pc}
t=T,…,N

Fig. 2.3 Overview for proposed SPLET control algorithm.

predict, as a heuristic, we randomly choose the public EV charging demand in one of the previous days to determine the day-ahead energy scheduling.

2.5.2 Stage One

In this stage, the decision on $\mathbf{p^a}$, the power to purchase from day-ahead market will be made. To calculate the expected cost, all possible realizations of the random parameters will be considered. For specific scenario o, the variables, e.g. the charging of employee EVs $\mathbf{p^c}$ and usage of time-of-use electricity $\mathbf{p^r}$, will be denoted as $\mathbf{p^{c,o}}$ and $\mathbf{p^{r,o}}$.

In this chapter, we use short-term forecasting to deal with the uncertainties of the base load power consumption and PV system output. In the first stage we need to predict these values for all the time slots in the next day. This means that if we use one hour as the duration for every time slot, we will need to implement predictions for all the twenty-four hours in the next day. There are different prediction methods for load forecasting. Here, we use Artificial Neural Network (ANN) [74] for the prediction of the base load power consumption, as it is shown to perform well on modeling a multivariable problem with complex nonlinear relationship [75, 76]. The neural network model and training method is based on [76], with one input layer, one hidden layer, and one output layer for the network. The weekday/weekend information for the next day, average ambient temperature of two previous days, and hourly power consumption of two previous days are used as input for the network. The output of the network is the predicted hourly power consumption for the next day.

Objective Function

In the first stage, the objective function is the operating cost for building EMS includes the transactional cost for day-ahead power market in the first stage and the expectation cost of the second stage, the latter including the cancellation fee for day-ahead power market and the real-time grid transaction cost for time-of-use electricity. Equation (2.4) shows the objective function, where $\mathbb{P}(o)$ denotes the probability for scenario o.

$$\min \ \ [f_1 := \sum_{t=1}^{N} I_t^a \cdot p_t^a + \sum_{o} \mathbb{P}(o) \cdot \sum_{t=1}^{N} (I_t^r \cdot p_t^{r,o} + I_t^p (p_t^a - p_t^{u,o}))] \tag{2.4}$$

If for time slot t, scenario o, the scheduled power in day-ahead power market p_t^a is larger than the actual used power $p_t^{u,o}$, the building EMS has to cancel the over-scheduled energy and pay a cancellation fee for that. When p_t^a is insufficient, the insufficient part will be provided by the time-of-use electricity. Besides the cost for purchasing in day-ahead market, the cost function also contains the expected cost of power usage from time-of-use electricity and the cancellation penalty for day-ahead market.

Constraints

To protect the power system and satisfy the power demand, the following constraints should all be satisfied.

Power balance equation: the power consumption should be balanced for every time slot under every scenario.

$$\forall t, \forall o, \hat{\mathcal{P}}_t^b + \sum_{i=1}^{M} p_{i,t}^{c,o} + p_t^{pc} = \mathcal{P}_t^{v,o} + p_t^{u,o} + p_t^{r,o} \tag{2.5}$$

The left side shows the power consumption including base load power consumption, employee EV charging power consumption, and public EV charging power consumption. The right side shows the power supplying which includes the PV output and used power from day-ahead power market and time-of-use electricity.

Day-ahead power limit: the actual used power from day-ahead market should be no more than the purchase.

$$\forall t, \forall o, \quad p_t^{u,o} \leq p_t^a \tag{2.6}$$

Charging power limit: the charging consumption for the employee EVs should be smaller than the maximal charging power defined by the EV specification.

$$\forall i, \forall t, \forall o, \quad 0 \leq p_{i,t}^{c,o} \leq p_i^{c,\max} \tag{2.7}$$

SOC requirement: Equation (2.8) describes that the EV battery SOC is increased with the amount of charged energy. Equation (2.9) shows the definition for the final EV SOC. Equation (2.10) requires that in the end, all EVs should get charged with a minimum required amount of energy. To protect the EV battery, we set a upper limit for EV SOC all the time. Equation (2.11) constrains that the EV SOC should be no larger than the upper limit ($\mathrm{S}^{\max}$) all times.

$$\forall i, \forall t, \forall o, \;\; s_{i,t}^o = s_{i,t-1}^o + \eta_i^e \cdot p_{i,t}^{c,o} \cdot L \cdot \frac{100}{E_i} \tag{2.8}$$

$$\forall i, \forall o, \quad s^{o}_{i,N} = SI^{o}_{i} + \sum_{t=1}^{N} \eta^{e}_{i} \cdot p^{c,o}_{i,t} \cdot L \cdot \frac{100}{E_i} \tag{2.9}$$

$$\forall i, \forall o, \quad s^{o}_{i,N} \geq S^{\min}_{i} \tag{2.10}$$

$$\forall i, \forall t, \forall o, \quad s^{o}_{i,t} \leq S^{\max}_{i} \tag{2.11}$$

Also, employees may leave before the expected departure time. Hence, the EV SOC should be maintained at a level higher than a lower bound, to support a certain driving range. Equation (2.12) constrains that the EV SOC should be larger than a lower bound ($\mathrm{S}^{\mathrm{minA}}_{i}$) after three hours of arrival time.

$$\forall i, \forall o, \forall (t \geq \mathcal{A}^{o}_{i} + 3), \quad s^{o}_{i,t} \geq S^{\mathrm{minA}}_{i} \tag{2.12}$$

The timing constraint: EVs can only be charged when they are parked in the building. This is implemented by using the auxiliary parameter $\mathcal{M}$, which is configured as zero if it is out of the time window between the EV arrival and departure. When EV i is parked in the building, $\mathcal{M}_{\mathbf{i}}$ will be equal to the maximal allowed charging rate of that EV.

$$\forall i, \forall t, \forall o, \quad 0 \leq p^{c,o}_{i,t} \leq \mathcal{M}^{o}_{i,t} \tag{2.13}$$

Scenarios Combination

The total number of scenarios for stochastic programming framework is related to the possible realization of uncertainties. Large number of scenarios will take a long computation time or even may not be schedulable. However, considering the employee EVs would arrive and leave the building around similar time slots, we can reduce the total number of scenarios with scenario combination for the computation of first stage for EVs with the same specifications.

For example, we assume there are five Honda EV fleets of 20 cars each for the office building. For simplification of demonstration, we only consider the uncertainties of arrival time and departure time. These EVs are assumed to arrive in the building either in the time slot of 7am - 8am or the time slot of 8am - 9am and will leave the building either in 5pm - 6pm or 6pm - 7pm. For every EV fleet, there are four possible realizations of the uncertainties. Without scenario reduction, there would be $5^4 = 625$ scenarios. However, we observe that for the purpose of day-ahead energy scheduling, it is unnecessary to know which EVs arrive/depart at a given time slot, as long as we know the number of EVs (since they are the same model). Hence, we can combine the scenarios as follows: for the arrival time, there are six possible realizations: five EV fleets arrive in the time slot of 7am - 8am; four EV fleets arrive in the time slot of 7am - 8am;...; zero EV fleet arrive in the time slot of 7am - 8am. And for the departure time, there are also six kinds of possible realizations. Then in total there are 36 possible scenarios. The total number of scenarios is significantly reduced with this kind of scenario combination. To point out, this kind of scenarios combination does not lose the computation accuracy as it only combine the scenarios and update the probability for every scenario accordingly.

EV V2B

It is assumed that V2B is implemented for employee EVs which acts as a countermeasure for the mismatch between day-ahead power scheduling and actual demand realization. When V2B technique is adopted, the EV battery degradation cost should be taken into consideration as part of the total operating cost. Minimization of the cost function f_1 described in Equation 2.4 is now revised as follows.

$$\min \ \ [f_2 := f_1 + \sum_{o} \mathbb{P}(o) \cdot \sum_{i=1}^{M} c_i^{e,o}] \tag{2.14}$$

where the EV battery degradation cost $c_i^{e,o}$ is calculated as in Equation (2.1).

When EV V2B is adopted, several constraints should be revised or added. Equation (2.8) is updated to include the effect of discharging on EV SOC. The final EV SOC is now defined in Equation (2.16).

$$\forall i, \forall t, \forall o, \ \ s_{i,t}^{o} = s_{i,t-1}^{o} + (\eta_{e,i} \cdot p_{i,t}^{c,o} - \frac{p_{i,t}^{d,o}}{\eta_{e,i}}) \cdot L \cdot \frac{100}{E_i} \tag{2.15}$$

$$\forall i, \forall o, \ \ \ s_{i,N}^{o} = SI_i^{o} + \sum_{t=1}^{N} (\eta_{e,i} \cdot p_{i,t}^{c,o} - \frac{p_{i,t}^{d,o}}{\eta_{e,i}}) \cdot L \cdot \frac{100}{E_i} \tag{2.16}$$

The new power balance equation is now shown in Equation (2.17).

$$\forall t, \forall o, \ \ \hat{\mathcal{P}}_t^{b,o} + \sum_{i=1}^{M} p_{i,t}^{c,o} + p_t^{pc} = \hat{\mathcal{P}}_t^{v} + p_t^{u,o} + p_t^{r,o} + \sum_{i=1}^{M} p_{i,t}^{d,o} \tag{2.17}$$

The discharging rate of EVs is constrained in Equation (2.18).

$$\forall i, \forall t, \forall o, \ \ 0 \leq p_{i,t}^{d,o} \leq P_i^{c,\max} \tag{2.18}$$

Stand-alone Battery System

With V2B technique, the energy stored in EVs could be discharged back to the building and help reduce the total operating cost. However, the EV V2B can only be used when the EVs are parked in the building and there is enough time for EVs to get charged with enough energy before the departure time. Here, we further consider that the building is integrated with a stand-alone lead-acid battery system. The battery system could also be used to mitigate the mismatch between day-ahead power scheduling and actual demand realization. The ageing cost of the stand-alone battery is considered as part of the total operating cost, hence the cost function is revised as in Equation (2.19).

$$\min[f_3 := f_2 + \sum_{o} \mathbb{P}(o) \cdot \sum_{t=1}^{N} c_t^{ba,o}] \tag{2.19}$$

When stand-alone battery system is adopted, two more constraints should be satisfied. The charging and discharging rates of battery system are constrained as in Equation (2.20).

$$\forall t, \forall o, \quad 0 \leq p_t^{ba,c,o}, p_t^{ba,d,o} \leq P^{ba,\max} \tag{2.20}$$

The updated power balance equation is shown as follows.

$$\begin{aligned} \forall t, \forall o, \quad \hat{\mathcal{P}}_t^{b,o} + \sum_{i=1}^{M} p_{i,t}^{c,o} + p_t^{ba,c,o} =& \hat{\mathcal{P}}_t^{v} + p_t^{u,o} + p_t^{r,o} \\ &+ \sum_{i=1}^{M} p_{i,t}^{d,o} + p_t^{ba,d,o} \end{aligned} \tag{2.21}$$

2.5.3 Stage Two

The decision on $\mathbf{p^a}$ (the power to purchase from day-ahead market) will be made at the end of the first stage. In stage two, we can use $\mathbf{p^a}$ as a constant input and the building EMS will schedule the employee EVs charging and coordinate with the time-of-use electricity. The charging demand of the public EVs will also be satisfied in this stage. Fig. 2.4 shows the flowchart for this stage.

The formulation in stage two is largely similar to that of stage one, with the following differences:

- The value of the purchased power from day-ahead market $\mathbf{p^a}$ is decided in the first stage and now can be regarded as a constant. This is related to Step 1 of Fig. 2.4.

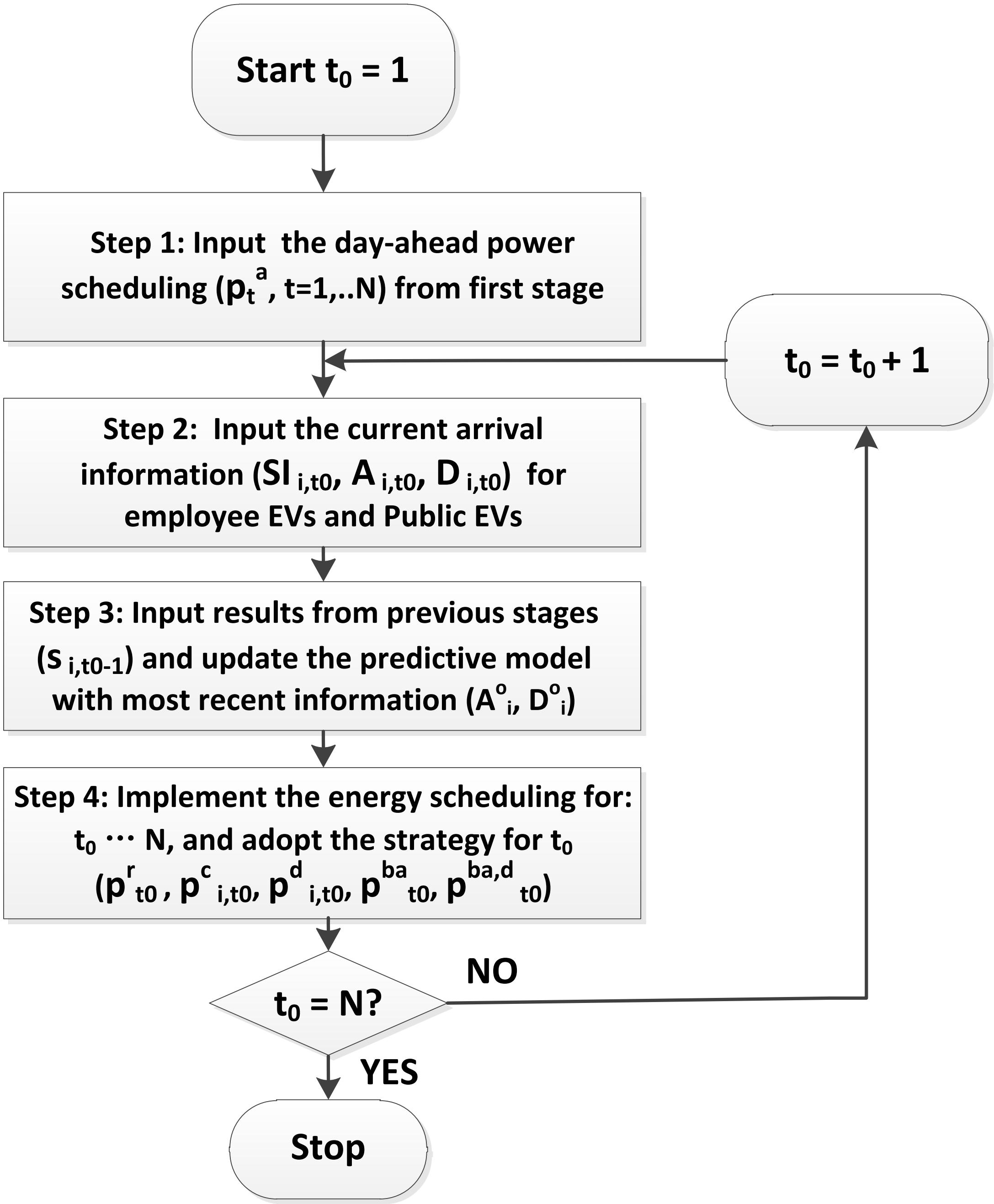
Start $t_0 = 1$
Step 1: Input the day-ahead power scheduling (p_t^a, t=1,..N) from first stage
$t_0 = t_0 + 1$
Step 2: Input the current arrival information ($SI_{i,t0}$, $A_{i,t0}$, $D_{i,t0}$) for employee EVs and Public EVs
Step 3: Input results from previous stages ($s_{i,t0-1}$) and update the predictive model with most recent information (A^o_i, D^o_i)
Step 4: Implement the energy scheduling for: $t_0 \cdots N$, and adopt the strategy for t_0 (p^r_{t0}, $p^c_{i,t0}$, $p^d_{i,t0}$, p^{ba}_{t0}, $p^{ba,d}_{t0}$)
$t_0 = N$?
NO
YES
Stop

Fig. 2.4 Flowchart for the second stage of SPLET.

- The set of possible scenarios and their distributions should be updated. With the new EV arrival/departure information, the probability distribution are updated for EV arrival time, arrival SOC, and departure time. For example, we assume the arrival time for EV i follows a distribution of $\mathbb{P}(7am - 8am) = 0.2$, $\mathbb{P}(8am - 9am) = 0.6$, $\mathbb{P}(9am - 10am) = 0.2$. If until 8am, it has not arrived, the probability distribution of the arrival time should be updated as $\mathbb{P}(8am - 9am) = 0.6/(1 - 0.2) = 0.75$, $\mathbb{P}(9am - 10am) = 0.25$. Otherwise, the probability should be $\mathbb{P}(8am - 9am) = \mathbb{P}(9am - 10am) = 0$. Also, the prediction for base load power consumption and PV output would be more accurate with more recent information. This is related to Step 2 and Step 3 of Fig. 2.4.

- For a specific time slot T, all the decision variables for previous time slots $1, ..., T-1$, have been decided already and now can be regarded as constants. The related decision variables for time slot T including p^r_T, $p^c_{i,T}$, $p^d_{i,T}$, $p^{ba,c}_{i,T}$, and $p^{ba,d}_{i,T}$, should be decided with realized actual demand for time slot T and existing decided variables. This is related to Step 4 of Fig. 2.4.

2.5.4 Sample Average Approximation based SPLET

As mentioned in Section 2.5.2, scenario combination technique can help reduce the number of scenarios to compute. However, the total number of all the possible scenarios will still increase very quickly with the increasing of the number of the EV fleets and possible realizations of uncertainties for one EV fleet. Here, SAA_SPLET is proposed which can use sample average approximation to further reduce the scenarios to compute.

Sample average approximation can help reduce the computation requirement for stochastic programming problems [77]. The difference between the two proposed algorithms, SPLET and SAA_SPLET, is that for SAA_SPLET, some sample scenarios will be chosen from all the possible scenarios and the objective function will be computed based on these chosen sample scenarios. Scenario combination proposed for SPLET is not used in SAA_SPLET. The methodology in the second stage for SPLET and SAA_SPLET are the same. There are different kinds of sampling methods, Monte Carlo sampling is used in this chapter. For every sample scenario, the variables with uncertainties will be chosen according to their distributions. For example, if we consider a variable has two possible values: A and B. Each value has the same realization probability: 0.5. Hence, for every sample scenario, we can generate a random number between 0 and 1, if the random number is larger than 0.5 we will choose A as the value for this variable, otherwise we will choose B for this variable. The number of samples will affect the computation time and estimation

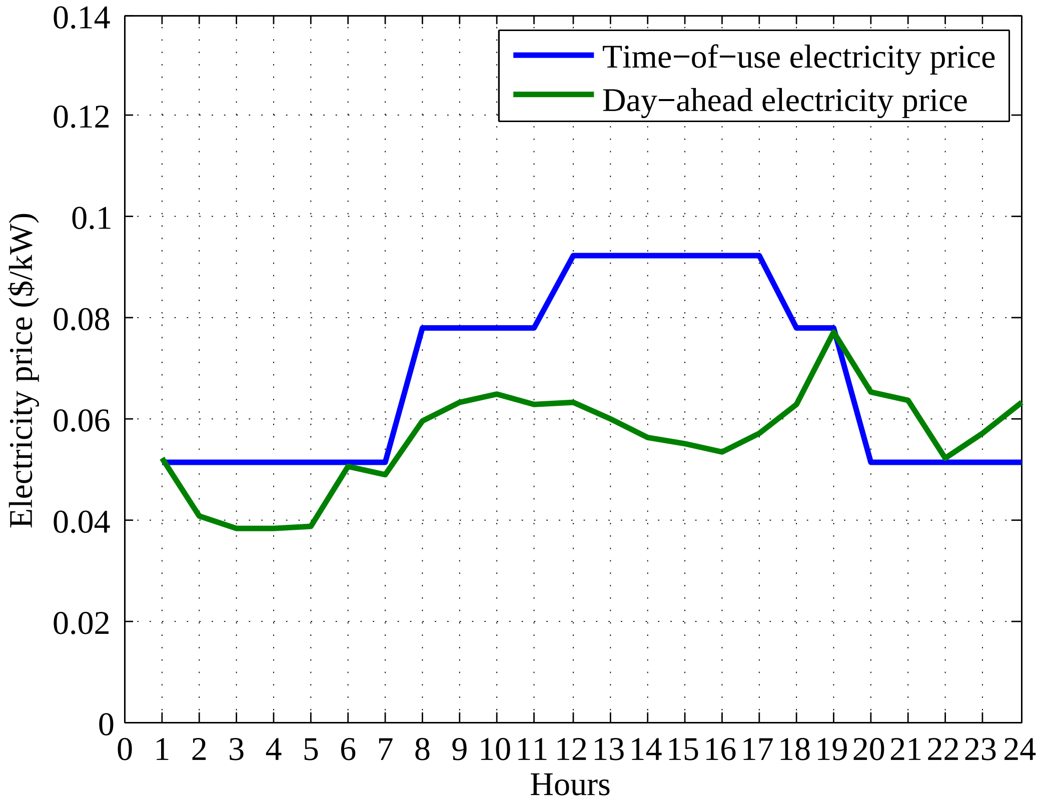

Fig. 2.5 Hourly time-of-use and day-ahead electricity price.

accuracy of SAA_SPLET. This will be discussed in Section 2.6.

2.6 Case Study and Discussion

In this section, the performance of the proposed algorithms are evaluated with real-world data. One hour is chosen as the duration for each time slot ($L = 1$ hour). The optimization problem is modeled in Java and solved by the IBM ILOG CPLEX Optimizer 12.0. All simulations are run on a desktop with an Intel i5 CPU and 32 GB memory.

The case study is based on the real-world power consumption of a large office building in Los-Angeles, obtained from Open EI [70]. It is a large office building and there are about 2000 employees in this building. Base load power consumption for two winter days (January 19th, 20th) and two summer days (July 19th, 20th) are chosen for simulation. The hourly day-ahead electricity price is obtained from ISO New England [78], as illustrated in Fig. 2.5.

This figure also shows the time-of-use price structure for evaluating the proposed control algorithms. The time-of-use price structure [79] is determined by the utility companies, which will be high for the peak power consumption hours of a day and low for the valley hours of a day. The day-ahead energy cancellation fee is set at a fixed value 0.027$/kWh.

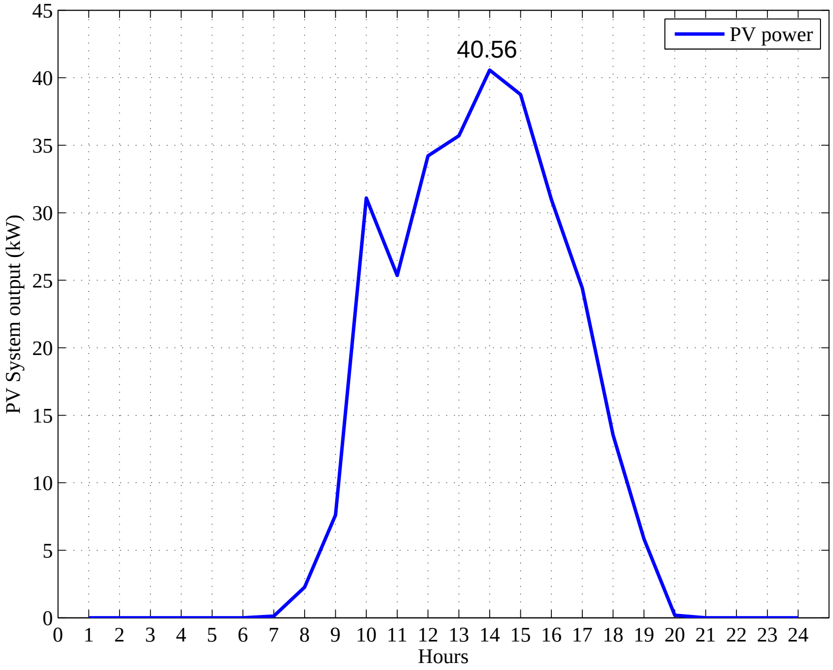

Fig. 2.6 PV system output.

The ANN described in Section 2.5.2 is used for the forecasting of the base load power consumption for the next day, which utilizes the weather information obtained from IEM [80]. The maximal base load forecasting error (mean square error) is 5.2% for the studied four days. Here, we consider there are 200 PV panels (each with 200W peak power generation) for the building. Due to the lack of radiation data for the building, the PV system outputs described in Section 2.4.4 are adopted for simulation with average irradiance data from [72] to get a PV system output for one day as shown in Fig. 2.6. We can see that the maximum hourly PV aggregated output is 40.56 kW. We use this average PV output data for the case study.

The EV specifications [67, 68] are shown in Table 2.1. The one-way EV battery plus charger conversion efficiency is 90%, and the EV battery degradation coefficient is -0.0027%. The stand-alone battery capacity is set to be 500 kWh, and the maximal charg-

Table 2.1 Electric vehicle specifications

Model	Charging power (kW)	Battery capacity (kWh)
Honda Fit	6.6	20
Tesla Model S	10	85

ing and discharging rate is 60 kW. Initial SOC for the stand-alone battery system is set to be 80. The lower bound for employee EV SOC in case of advance departure is set as 30. The required departure EV SOC is set as 90 and maximal allowed EV SOC is set as 100 for employee EVs. It is assumed that between 8am and 8pm, for every hour, at most twenty public EVs could get charging service from the office building. The public EVs charging demand is treated as an aggregated value $\mathbf{p}^{\mathbf{pc}}$. For one specific time slot, the number of public EVs that require charging service is treated as a random number ranging between 0 and 20. The public EV could be either Honda Fit or Tesla Model S with equal probability. The public EVs are assumed to be charged with moderate charging, which means that they will be charged with rated charging rate. In this Section, we first analyze the computation time and estimation accuracy for the proposed algorithms and then study the effectiveness of the proposed algorithms on operating cost reduction.

2.6.1 Computation Time and Estimation Accuracy Analysis for SPLET and SAA_SPLET

Here, we compare the computation time and estimation accuracy of the proposed two algorithms: SPLET, SAA_SPLET with Full_Scenario_SPLET which means that we do not use the scenario combination technique proposed in SPLET and compute the objective function with all the possible scenarios directly. Considering the schedulability of Full_Scenario_SPLET, only a small system with uncertainties in arrival SOC is studied. It is assumed that there are 10 fleets of EVs, each with 20 EVs and the arrival SOC could be either 40 or 60 with equal probability. Then there would be in total $2^{10} = 1024$ scenarios for Full_Scenario_SPLET. For SPLET, there would be 11 scenarios: scenario 1 means all the ten fleets arrive with SOC of 40 while scenario 11 means all the ten fleets arrive with SOC of 60. Base load power consumption for one winter day is used for this comparison.

For SAA_SPLET, we choose two kinds of sample size: 50 and 100. Twenty repetitions are run for each sample size to get the average result. Table 2.2 shows the scenario number,

Table 2.2 Computation and estimation analysis for SPLET and SAA_SPLET

Algorithm	Scenarios	Time (s)	operating cost ($)
Full_Scenario_SPLET	1024	11.83	3048.34
SPLET	11	0.42	3048.34
SAA_SPLET (50 Samples)	50	0.97	3087.75
SAA_SPLET (100 Samples)	100	1.63	3077.18

average computation time and operating cost for different control algorithms. We can see that SPLET could provide exactly the same result as Full_Scenario_SPLET while reducing 96.4% of the computation time. The proposed SAA_SPLET could also reduce the computation time significantly (91.8% for 50 samples, and 86.2% for 100 samples) while the average operating cost is a little higher than the Full_Scenario_SPLET (1.3% for 50 samples, and 0.95% for 100 samples). In this comparison, the computation time for SPLET is lower than SAA_SPLET with 50 samples and 100 samples. However the computation complexity of SPLET will increase very fast when more kinds of uncertainties are considered. For example, when we further consider the uncertainties for arrival time and departure time and each with two possible realizations, there would be in total 11*11*11 = 1331 scenarios for SPLET. Under this condition, the average computation time for SPLET is 768.21 seconds. The average computation time for SAA_SPLET with 100 samples is 2.27 seconds, which is much more computationally efficient compared with SPLET. For this setting, the Full_Scenario_SPLET is clearly infeasible.

2.6.2 Building operating cost Analysis for SPLET and SAA_SPLET

To demonstrate the effectiveness of the proposed control algorithms, the proposed two control algorithms are compared with two baseline algorithms. The first baseline control algorithm is As Soon As Possible (ASAP) charging which means that the building will only use the time-of-use electricity and charge both the employee EVs and public EVs as soon as possible after their arrival. The second baseline is the control algorithm proposed in [40] which is an EV charging control algorithm for a smart building based on the prediction of PV system output and base load power consumption. We denote the second baseline control algorithm as PVPB which stands for control with prediction of PV system output and Power consumption for Base load. In PVPB, the uncertainties of EV charging demand is not

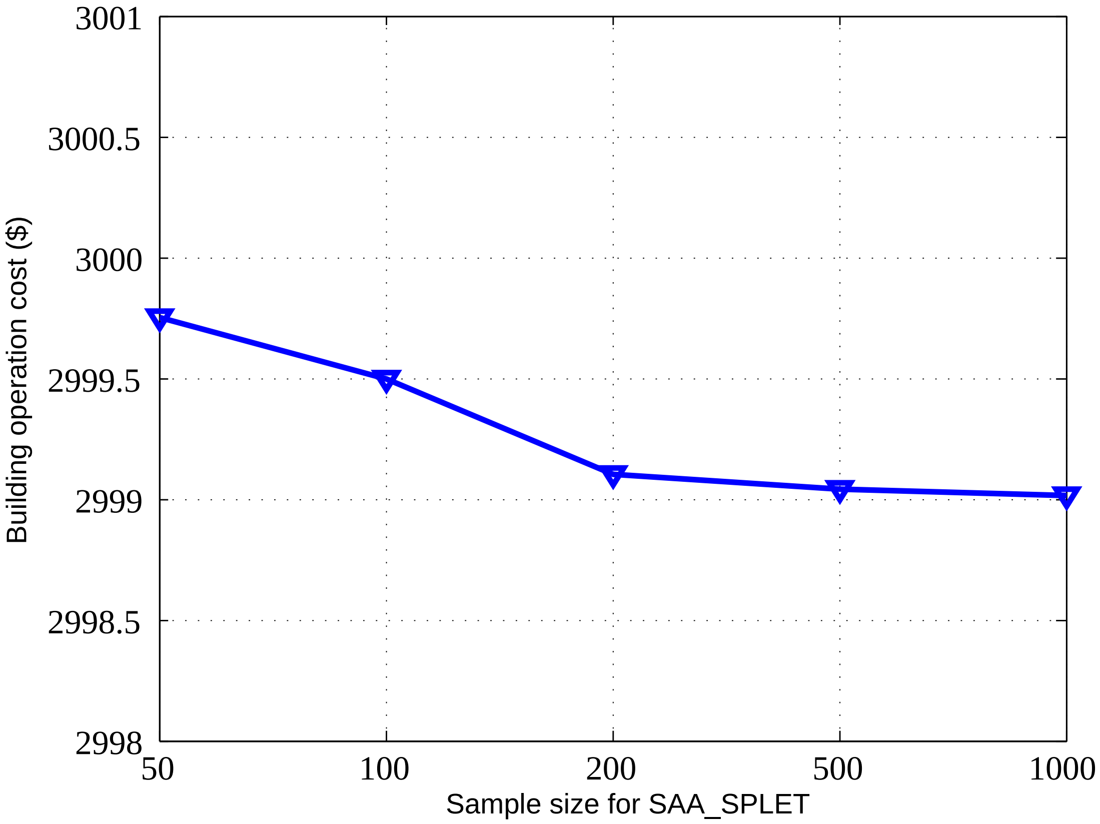

Fig. 2.7 Building operating cost with sample size.

considered, and assuming that the building EMS can have full information of EV charging demand including the arrival time, departure time and energy demand information.

The employee EVs are set as: 10 fleets and each fleet with 20 EVs. Each EV fleet has its independent arrival time, arrival SOC, and departure time. The EV arrival time distribution is uniformly distributed between 7am to 9am. The EV departure time distribution is uniformly distributed between 5pm to 7pm. The possible arrival SOC is set as 40, 50, 60 with equal probability. To determine the appropriate sample size, a performance comparison is presented with different sample sizes. Fig. 2.7 shows the average operating cost for one winter day with different sample sizes of SAA_SPLET. It can be seen that the operating costs are very close for different sample sizes. Here, 500 is chosen as the sample size for the following comparison. The public EVs are assumed to be charged with moderate charging.

As a comparative index, we calculate the operating cost reduction for the proposed two

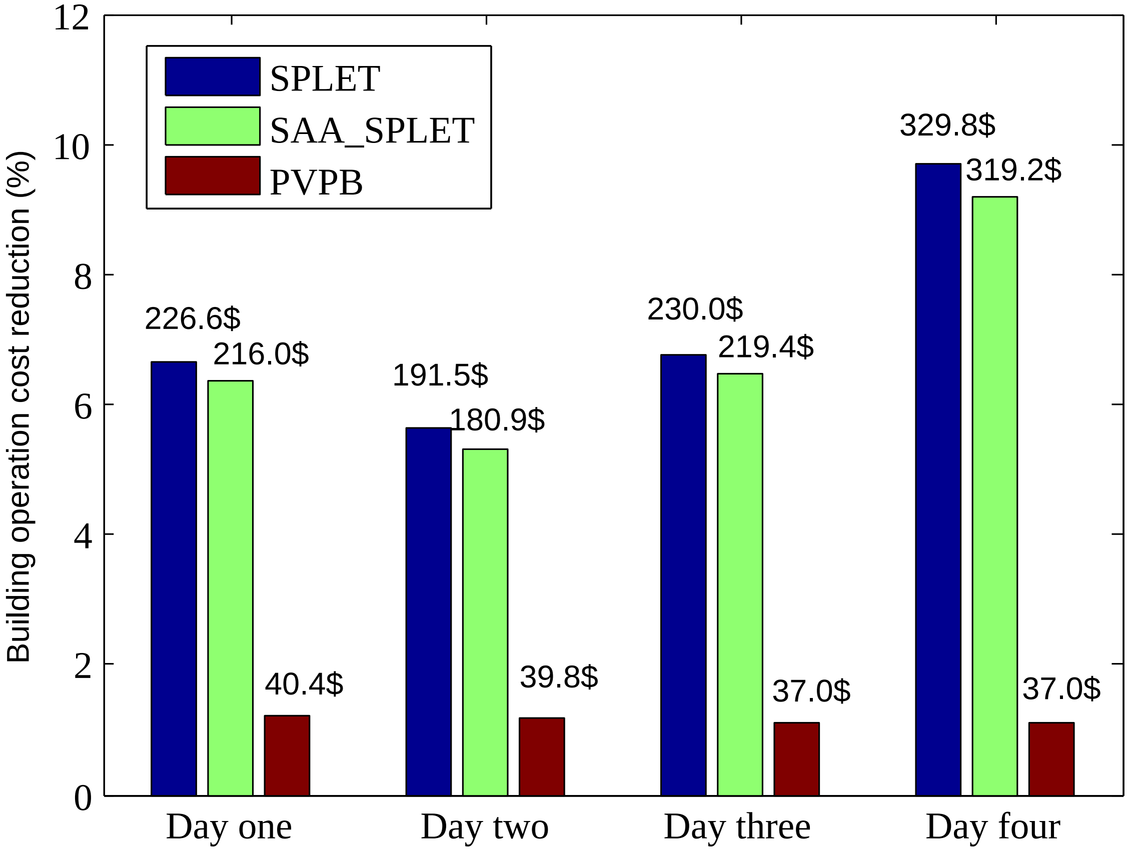

Fig. 2.8 Building operating cost reduction for two proposed algorithms.

control algorithms and PVPB over the first baseline control algorithm ASAP. The percentage cost reduction and value cost reductions compared with ASAP control for four days are shown in Fig. 2.8. We can see that for both SPLET and SAA_SPLET, the building operating cost can be significantly reduced compared with the baseline. For SPLET, the average daily percentage cost reduction is 7.2% and the average value cost reduction is \$244.5 (USD). For SAA_SPLET, the average daily percentage cost reduction is 6.9% and the average daily value cost reduction is \$233.9 (USD) . Both the SPLET and SAA_SPLET significantly outperform PVPB control which has the average 1.1% percentage cost reduction and 38.5 \$ value cost reduction. With the proposed control algorithms, the charging scheduling for EV fleets could be determined in the second stage.

Fig. 2.9 shows total power consumption for July 20th for the building with SPLET control and ASAP charging. In Fig. 2.9, PT means total power consumption from the

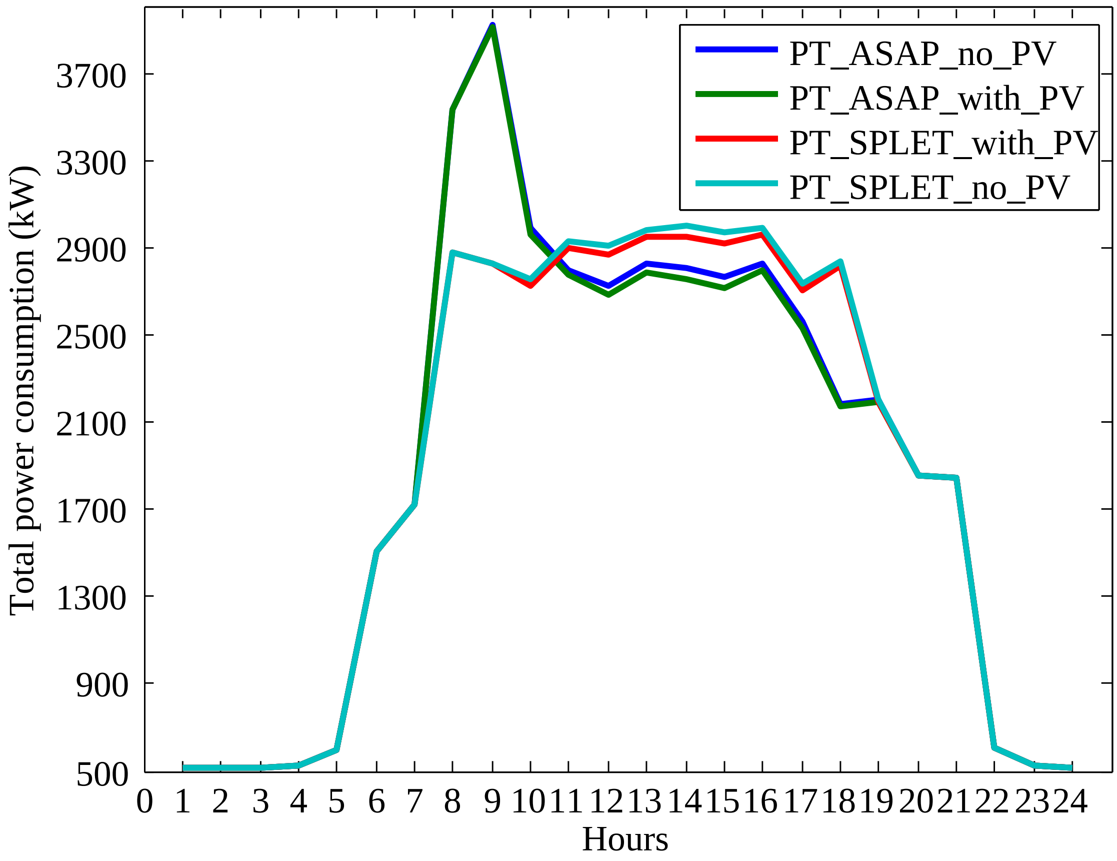

Fig. 2.9 Total electricity power consumption for building.

grid for the building. We can see that when we use the SPLET control algorithm, the peak power consumption is significantly reduced. This is because that with SPLET, the EV charging demand is well distributed in the whole day. Also, with the adoption of PV system, the power consumption from the grid is reduced for the daytime when there is positive output from the PV system. Fig. 2.10 shows the EV charging scheduling for two EV fleets under SPLET control. We can see that the EV charging power consumption is well distributed in all time slots.

2.6.3 Building with EV V2B and Stand-alone Battery System

Since the employee EVs could be treated as a kind of flexible load, V2B for employee EVs can be used to mitigate the mismatch between the day-ahead power scheduling and actual realization of power demand. Fig. 2.11 shows the charging and discharging rate for two

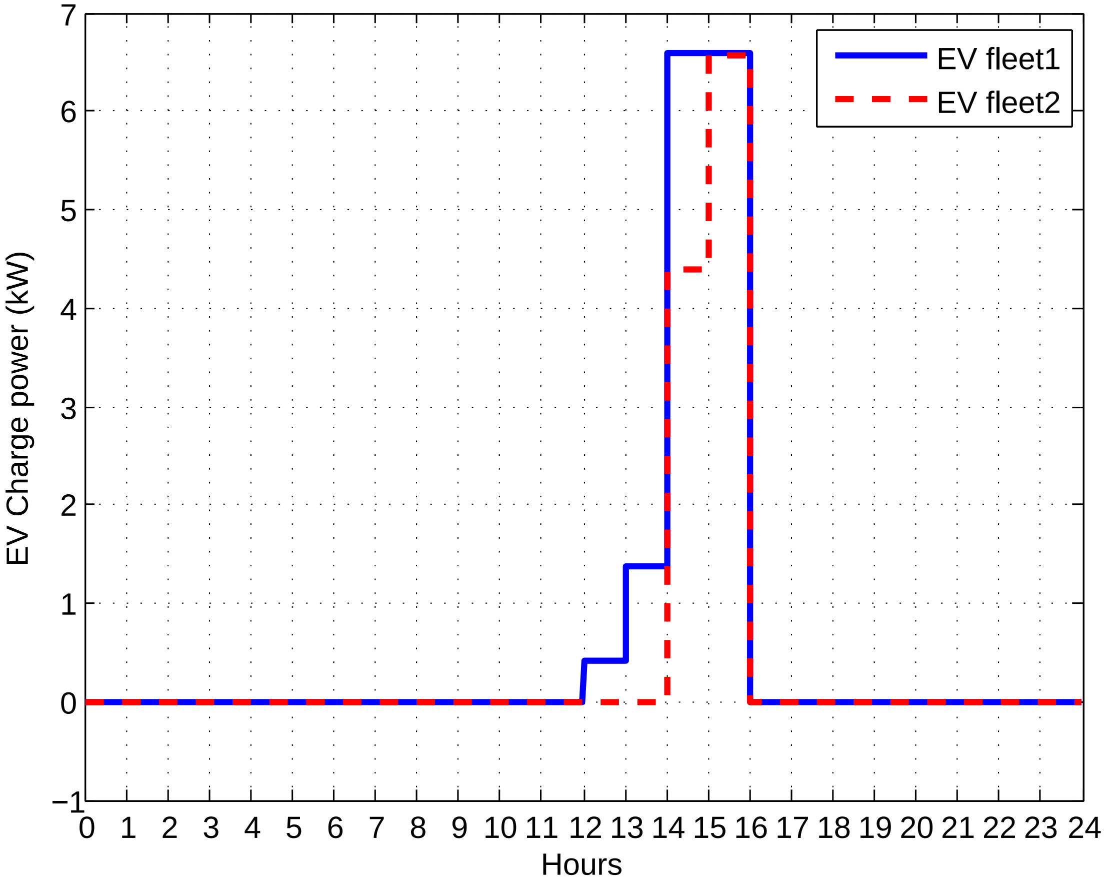

Fig. 2.10 Charging scheduling for EVs.

EV fleets for July 20th. We can see that for time slot 8am, and 1pm, the energy stored in the EV battery is discharged back to the building.

Table 2.3 shows the average total operating cost reduction and expected annual cost reduction for the building with and without EV V2B or with and without stand-alone battery system under SPLET control algorithm. For all the four days, both the V2B and stand-alone battery could help further reduce the building operating cost (22.06 $ and 22.67$). When the two technologies are adopted together, we could get the most operating cost reduction (30.74$). We can see that the expected annual cost reduction will be as high as 11220.1$ (assuming 365-day year), when both of the two technologies are adopted. As EV battery ageing cost is already taken into consideration as part of the total operating cost, the cost reduction can be seen as net benefit for the building.

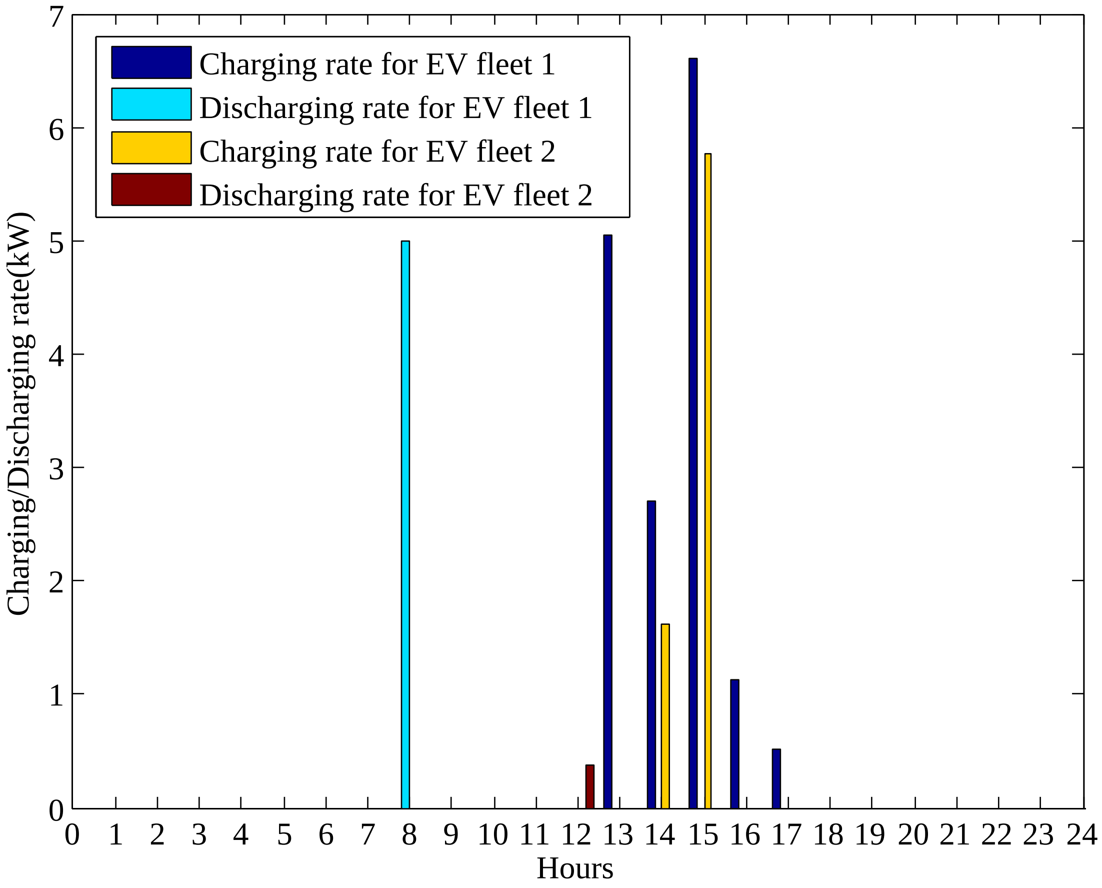

Fig. 2.11 Charging and discharging rate for employee EVs.

Fig. 2.12 shows the charging rate, discharging rate and changing of battery SOC for the stand-alone battery system for one summer day (July 20th). Table 2.4 shows the exact value for battery SOC for the stand-alone battery system. We can see that the energy stored in the stand-alone battery system is discharged to the building continuously from 7am to 7pm. The stand-alone battery system is recharged in the evening when the time-of-use electricity price is low. The discharging of stand-alone battery system can shelp reduce the peak power consumption and save the total building operating cost.

Table 2.3 Daily and annual operating cost reduction

Method	Daily Further Cost Reduction ($)	Annual Cost Reduction ($)
V2B	22.06	8051.9
SAB	22.67	8274.55
V2B + SAB	30.74	11220.1

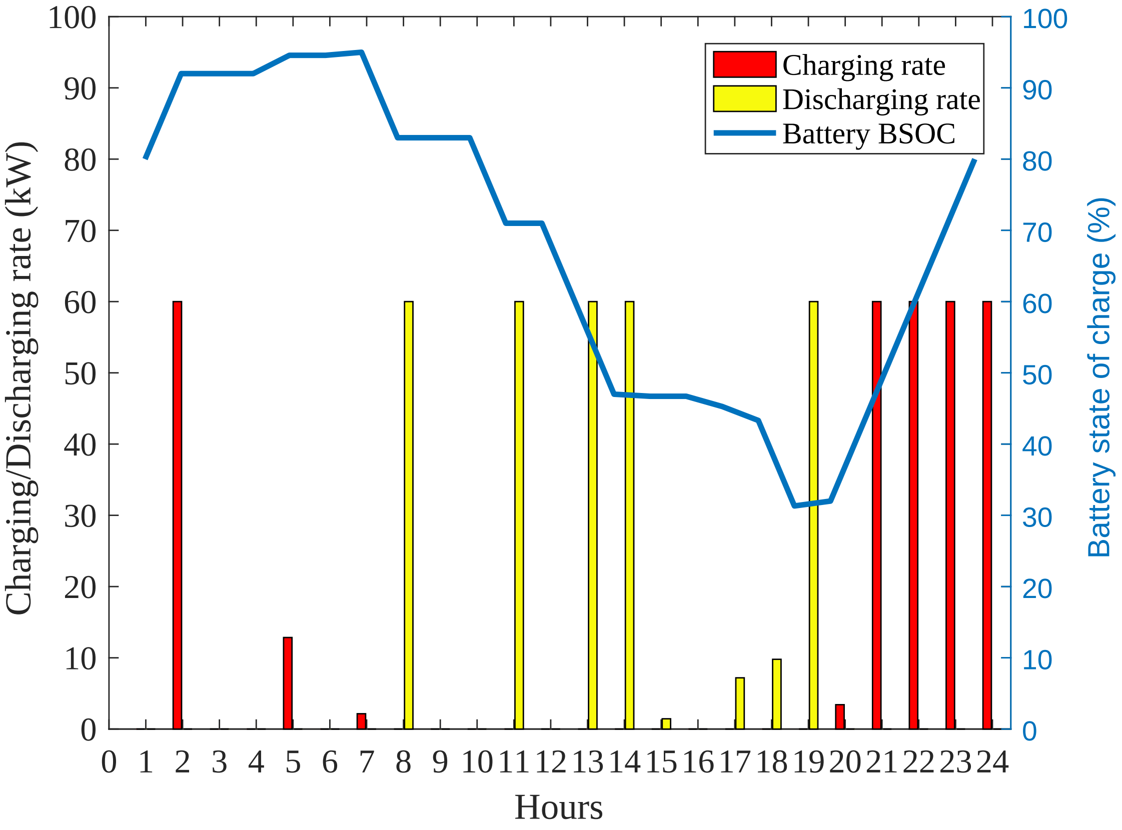

Fig. 2.12 Charging, discharging rate and SOC for stand-alone battery system.

2.7 Chapter Summary

Workplace EV charging is now supported by more and more companies to promote transportation electrification. In this chapter, we address the challenges for workplace EV charging and propose two computationally efficient scheduling algorithms: SPLET and SAA_SPLET. The proposed algorithms operate in two stages: day-ahead scheduling and

Table 2.4 Stand-alone battery system BSOC

Hour	1	2	3	4	5	6	7	8
BSOC	80	92	92	92	94.57	94.57	95	83
Hour	9	10	11	12	13	14	15	16
BSOC	83	83	71	71	59	47	46.71	46.71
Hour	17	18	19	20	21	22	23	24
BSOC	45.27	43.31	31.32	32	44	56	68	80

real-time operation. Simulation results with real-world data demonstrate the efficiency and effectiveness of the proposed algorithms. First, they can provide significant reduction for the building operating cost: 7.2% for SPLET and 6.9% for SAA_SPLET. Second, the proposed algorithms are computationally efficient and are suitable for real-time operation. Third, EV V2B and stand-alone battery system can be used as countermeasures for the mismatch between the actual realization of power demand and day-ahead scheduling. It is shown that the total operating cost of the building could be further reduced when both techniques are adopted.

In this chapter, we focus on tackling the challenges for energy management of office buildings and propose two energy control algorithms with stochastic programming framework and short-term electricity load forecasting. With the advancement of computation tools and advanced meters, energy management for residential homes now drawing more attention. In following two chapters, we demonstrate that smart homes can also benefit from machine learning algorithms.

Chapter 3

Boosting Based Multiple Kernel Learning and Transfer Regression for Electricity Load Forecasting

3.1 Introduction

Electricity load forecasting is very important for the secure and economic operation of a power system. The accuracy of electricity load forecasting directly influences the planning and control of power system operation. It is estimated that a 1% increase of forecasting error would bring in a 10 million pounds increase in operating cost per year (in 1984) for the UK power system [81]. Experts believe that this effect could become even stronger, due to the emergence of highly uncertain energy sources, such as solar and wind energy generation. Depending on the lead time horizon, electricity load forecasting ranges from short-term forecasting (minutes or hours ahead) which is useful for real-time control and power generation dispatch optimization, to long-term forecasting (years ahead) mainly for planning [75]. With increasingly competitive markets and demand response energy management [36], short-term load forecasting is becoming more and more important [82]. In this chapter, therefore, we focus on tackling this problem.

Electricity load forecasting is a difficult task since the load is influenced by many uncertain factors. Historical electricity load consumption, weather conditions, season effect and social activities are all possible influencing factors for the electricity usage. Various methods have been proposed for electricity load forecasting including statistical methods, time series analysis, and machine learning algorithms [83]. In [75], the authors review the implementation of neural networks for electricity load forecasting. The performance of general additive models for electricity load forecasting is discussed in [84].

Kernel methods applied on time series data is used to find a good kernel similarity to distinguish between time series [85]. Recent work uses multiple kernels to build prediction models for electricity load forecasting. For example, in [86], Gaussian kernels with different parameters are applied to learn peak power consumption. In [87], different types of kernels are used for different features and a multi-task learning algorithm is proposed and applied on low level load consumption data to improve the aggregated load forecasting accuracy. However, all of the existing methods rely on a fixed set of coefficients for the kernels (i.e., simply set to 1), implicitly assuming that all the kernels are equally important for forecasting, which may be suboptimal in real-world applications.

Multiple kernel learning (MKL) [88], which learns both the kernels and their combination weights for different kernels, could be tailored to this problem. Through MKL, different kernels could have different weights according to their influence on the outputs. However, learning with multiple kernels usually involves a complicated convex optimization problem, which limits their application on large scale problems. Although some progress has been made in improving the efficiency of the learning algorithms, most of them only focus on classification tasks [89, 90]. On the other hand, electricity load forecasting is a regression problem and the computation time is an important issue.

Another practical issue for load forecasting is the lack of data to build a reliable forecasting model. For example, consider the case of a set of newly built houses (target domain) for which we want to predict the load consumption. We may not have enough data to build a prediction model for these new houses, while we have a large amount of data or knowledge from other houses (source domain). The challenge here is to perform transfer learning [91], which relies on the assumption that there are some common structures or factors that can be shared across the domains. The objective of transfer learning for load forecasting is to improve the forecasting performance by discovering shared knowledge and leveraging it for electricity load prediction for target buildings.

Various techniques have been proposed to efficiently learn MKL models [92], and our BMKR algorithm is originally inspired by [90], which applies the idea of AdaBoost [93] to train a multiple kernel based classifier. BMKR is a more general framework which can adopt different loss functions for different learning tasks. Furthermore, the boosting approach provides a natural approach to solve small sample size problems by leveraging transfer learning techniques. The original work on boosting based transfer learning proposed in [94] introduces a sample-reweighting mechanism based on AdaBoost for classification problem. Later, this approach is generalized to the cases of regression [95], and transferring knowledge from multiple sources [96]. In [97], a gradient boosting based algorithm is proposed for multi-task learning, where the assumption is that the model parameters of all the tasks share a common factor. In [98], the transfer boosting and multi-task boosting algorithms are generalized to the context of online learning. While both multiple kernel learning and transfer learning have been studied extensively, the effort taking advantage of transfer learning to solve multiple kernel learning problem is very limited. Our BTMKR algorithm distinguishes itself from these methods because it deals with these two learning problems in a unified and principled approach. To our best knowledge, this is the first attempt to transfer MKL for regression problem.

In this chapter, we address both challenges within a novel boosting-based MKL framework. In particular, we first propose the Boosting based Multiple Kernel Regression (BMKR) algorithm to improve the computational efficiency of MKL. Furthermore, we extend BMKR to the context of transfer learning, and propose two variants of Boosting based Transfer Multiple Kernel Regression (BTMKR): kernel-level boosting based transfer multiple kernel regression (K-BTMKR) and model-level gradient boosting based transfer multiple kernel regression (M-BTMKR). Our contribution, from an algorithmic perspective, is two-fold: We propose a boosting based learning framework 1. to learn regression models with multiple kernels efficiently, and 2. to leverage the MKL models learned from other domains. On the application side, this work introduces the use of transfer learning for the electricity load forecasting problem, which opens up potential future work avenues.

The rest of this chapter is organized as follows. The background for kernel based learning and boosting methods are discussed in Section 3.2. In Section 3.3, the proposed algorithms in this chapter are discussed. We first introduce the details for BMKR and then the two variants of BTMKR are discussed. Section 3.4 presents the simulation results for the proposed algorithms on short-term electricity load forecasting (one-step forward). Finally, the chapter summary is presented in Section 3.5.

3.2 Technique Background for Multiple Kernel Regression and Boosting Transfer

3.2.1 Multiple Kernel Regression

Let $\mathcal{S} = \{(x_n, y_n), n = 1, \ldots, N\} \in \mathbb{R}^d \times \mathbb{R}$ be the data set with N samples, $\mathcal{K} = \{k_m : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}, m = 1, \ldots, M\}$ be M kernel functions. The objective of MKL is to learn a prediction model, which is a linear combination of M kernels, by solving the following optimization problem [92]:

$$\min_{\eta \in \Delta} \min_{F \in \mathcal{H}_K} \frac{1}{2} ||F||_K^2 + C \sum_{n=1}^{N} \ell(F(x_n), y_n), \tag{3.1}$$

where $\Delta = \{\eta \in \mathbb{R}_+ | \sum_{m=1}^{M} \eta_m = 1\}$ is a set of weights, $\mathcal{H}_K$ is the reproducing kernel Hilbert space (RKHS) induced by the kernel $K(x, x_n) = \sum_{m=1}^{M} \eta_m k_m(x, x_n)$ and $\ell(F(x), y)$ is a loss function. In this chapter we use the squared loss $\ell(F(x), y) = \frac{1}{2}(F(x) - y)^2$ for the regression problem. The solution of Eq. 3.1 is of the form[1]

$$F(x) = \sum_{n=1}^{N} \alpha_n K(x, x_n), \tag{3.2}$$

where the coefficients $\{\alpha_n\}$ and $\{\eta_m\}$ are learned from samples.

Compared with single kernel approaches, MKL algorithms can provide better learning capability and alleviate the burden of designing specific kernels to handle diverse multivariate data.

3.2.2 Gradient Boosting and ϵ-Boosting

Gradient boosting [99, 100] is an ensemble learning framework which combines multiple hypotheses by performing gradient descent in function space. More specifically, the model learned by gradient boosting can be expressed as:

$$F(x) = \sum_{t=1}^{T} \rho^t f^t(x), \tag{3.3}$$

---

[1] We ignore the bias term for simplicity of analysis, but in practice, the regression function can accomodate both the kernel functions and the bias term.

where T is the number of total boosting iterations, and the t-th base learner f^t is selected such that the distance between f^t and the negative gradient of the loss function at $F = F^{t-1}$ is minimized:

$$f^t = \arg\min_{f} \sum_{n=1}^{N} \left(f(x_n) - r_n^t \right)^2 , \tag{3.4}$$

where $r_n^t = -\left[\frac{\partial \ell(F(x_n), y_n)}{\partial F}\right]_{F=F^{t-1}}$, and ρ^t is the step size which can either be fixed or chosen by line search. Plugging in the squared loss we have $r_n^t = y_n - F^{t-1}(x_n)$. In other words, gradient boosting with squared loss essentially fits the residual at each iteration.

Let $\mathcal{F} = \{f_1, \ldots, f_J\}$ be a set of candidate functions, where $J = |\mathcal{F}|$ is the size of the function space, and $f : \mathbb{R}^d \to \mathbb{R}^J$, $f(x) = [f_1(x), \ldots, f_J(x)]^\top$ be the mapping defined by $\mathcal{F}$. Gradient boosting with squared loss usually proceeds in a greedy way: the step size is simply set $\rho^t = 1$ for all iterations. On the other hand, if the step size ρ^t is set to some small constant $\epsilon > 0$, it can be shown that under the monotonicity condition, this example of gradient boosting algorithm, referred to as ϵ-boosting in [101], essentially solves an ℓ_1-regularized learning problem [102]:

$$\min_{||\beta||_1 \le \mu} \sum_{n=1}^{N} \frac{1}{N} \ell \left(\beta^\top f(x_n), y_n \right) , \tag{3.5}$$

where $\beta \in \mathbb{R}^J$ is the coefficient vector, and μ is the regularization parameter, such that $\epsilon T \le \mu$. In other words, ϵ-boosting implicitly controls the regularization via the number of iterations T rather than μ.

3.2.3 Transfer Learning Concepts

The training data and testing data are drawn from the same distribution is a common assumption for many machine learning algorithms [103]. However, for some real-world scenarios, this assumption does not hold. It could be very expensive or difficult to collect training data for the domain that we are interested in, referred to as target domain. Meanwhile, a large amount of data is available for some related domains referred to as source domains. Then it will be desirable if some knowledge could be transferred from source domains to target domain. This is the motivation for transfer learning.

The notations and definitions in this section match those defined in [91]. A domain

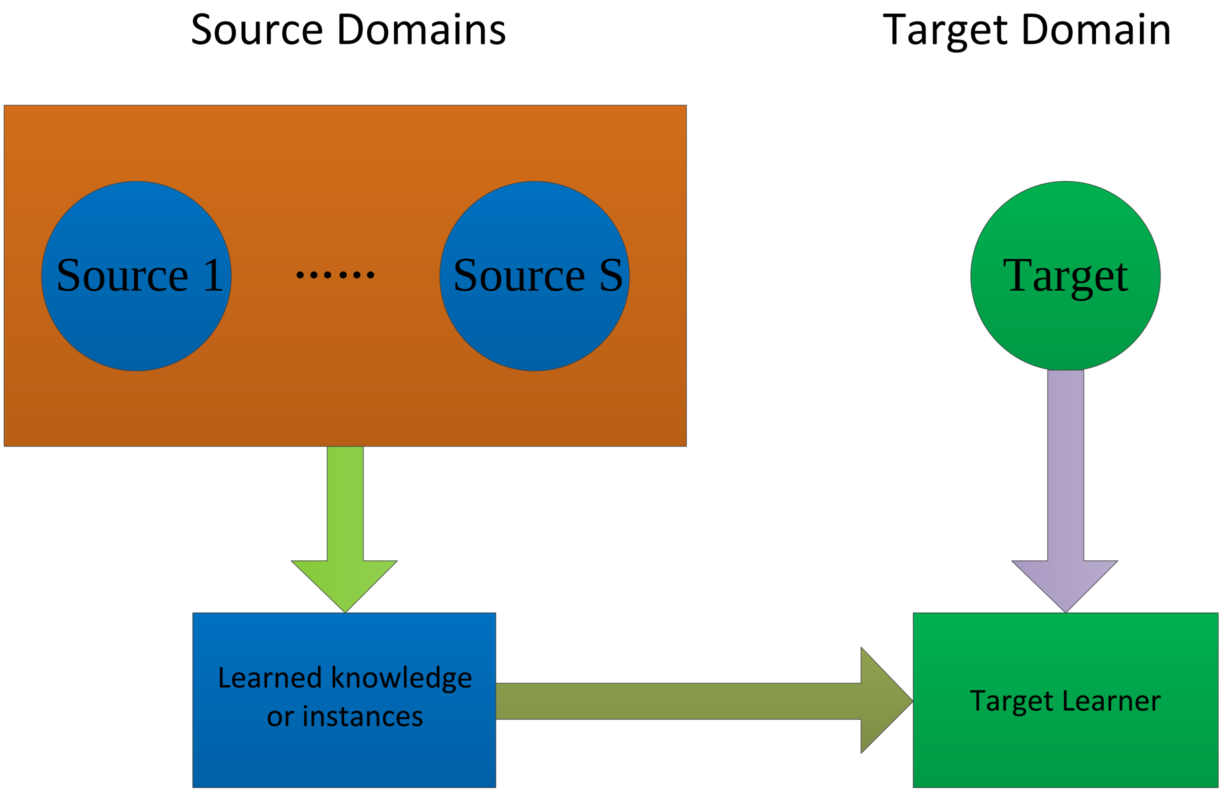

Fig. 3.1 Transfer learning process paradigm

$\mathcal{D}$ is defined by two components: feature space $\mathcal{X}$ and marginal probability $P(X)$, where $X = \{x_1, \ldots, x_n\} \in \mathcal{X}$. A task for a specific domain $\mathcal{D}$ is defined by two components: a label space $\mathcal{Y}$, and a predictive function $f(\cdot)$. Task $\mathcal{T}$ is denoted as $\mathcal{T} = \{\mathcal{Y}, f(\cdot)\}$. Transfer learning aims to improve the learning performance of the target predictive function in $\mathcal{D}_T$ using the knowledge in $\mathcal{D}_T$ and $\{\mathcal{D}_1, \ldots, \mathcal{D}_S\}$, where $\mathcal{D}_T$ is the target domain, and $\{\mathcal{D}_1, \ldots, \mathcal{D}_S\}$ are source domains. In this definition, we have S source domains $\mathcal{D}_1, \ldots, \mathcal{D}_S$ and one target domain $\mathcal{D}_T$. Figure 3.1 shows the basic learning paradigm for transfer learning.

Transfer learning aims to improve the learning performances of the predictive model in target domain by using knowledge learned from one or more related source domains. However, when the source domains are not closely related to the target domain, the knowledge learned from the source domains may impose a detrimental effect on the target learner. This effect is referred to as negative transfer. As presented in [104], source domain needs to be sufficiently related to target domain, otherwise transfer learning will not work. Thus, we need to pay attention to potential negative transfer during the transfer learning process.

3.2.4 Transfer Learning from Multiple Sources

Let $\mathcal{S}_T = \{(x_n, y_n), n = 1, \dots, N\}$ be the data set from the target domain, and $\{\mathcal{S}_1, \dots, \mathcal{S}_S\}$ be the data sets from S source domains, where $\mathcal{S}_s = \{(x_n^s, y_n^s), n = 1, \dots, N_s\}$ are the samples of the s-th source. Let $\{F_1, \dots, F_S\}$ be the prediction models learned from S source domains. In this work, the s-th model F_s is trained by an MKL algorithm (e.g., BMKR), and is of the form:

$$F_s = \sum_{m=1}^{M} \eta_m^s h_m^s(x) = \sum_{m=1}^{M} \eta_m^s \sum_{n=1}^{N_s} \alpha_n^s k_m(x, x_n^s). \tag{3.6}$$

The objective of transfer learning is to build a model F that has a good generalization ability in the target domain using the data set $\mathcal{S}_T$ (which is typically small) and knowledge learned from sources $\{\mathcal{S}_1, \dots, \mathcal{S}_S\}$. In this work, we assume that such knowledge has been embedded into $\{F_1, \dots, F_S\}$, and therefore the problem becomes to explore the model structures that can be transferred to the target domain from various source domains. This type of learning approach is also referred to as parameter transfer [91].

3.3 Methods

3.3.1 Boosting based Multiple Kernel Learning Regression

The idea of BMKR is to learn an ensemble model with multiple kernel regressors using the gradient boosting framework. The starting point of our method is similar to multiple kernel boosting (MKBoost) [90], which adapts AdaBoost for multiple kernel classification. We extend this idea to a more general framework of gradient boosting [99, 100], which allows different loss functions for different types of learning problems. In this chapter, we focus on the regression problem and use the squared loss.

At the t-th boosting iteration, for each kernel $k_m, m = 1, \dots, M$, we first train a kernel regression model such as support vector regression (SVR) by fitting the current residuals,

Algorithm 1: BMKR: Boosting based Multiple Kernel Regression

```
Input: Data set $\mathcal{S}$, kernel functions $\mathcal{K}$, number of iterations $T$

Initialize residual: $r_n^1 = y_i, \forall n \in \{1, \ldots, N\}$, and $F = 0$
for $t = 1, ..., T$ do
    for $m = 1, ..., M$ do
        Sample $N'$ data points from $\mathcal{S}$
        Train a kernel regression model $f_m^t$ with $k_m$ by fitting the residuals of the selected $N'$
        samples
        Compute the loss: $e_m^t = \frac{1}{2}\sum_{n=1}^{N}\left(f_m^t(x_n) - r_n^t\right)^2$
    end for
    Select the regression model with the smallest fitting error: $f^t = \arg\min_{f_m^t} e_m^t$
    Add $f^t$ to the ensemble: $F \leftarrow F + \epsilon f^t$
    Update residuals: $r_n^{t+1} = y_n - F(x_n), \forall n \in \{1, 2, ...N\}$
end for
Output: the final multiple kernel function $F(x)$
```

and obtain a solution of the form:

$$f_m^t(x) = \sum_{n=1}^{N} \alpha_{t,n} k_m(x, x_n). \tag{3.7}$$

Then we choose from M candidates, the regression model with the smallest fitting error

$$f^t = \underset{f_m^t, m \in \{1, \ldots, M\}}{\arg\min} e_m^t, \tag{3.8}$$

where $e_m^t = \frac{1}{2}\sum_{n=1}^{N}\left(f_m^t(x_n) - r_n^t\right)^2$, and add it to the ensemble F. The final hypothesis of BMKR is expressed as in Eq. 3.3.

The pseudo-code of BMKR is shown in Algorithm 1. For gradient boosting with squared loss, the step size ρ^t is not strictly necessary [105], and we can either simply set it to 1, or a fixed small value ϵ as suggested by ϵ-boosting. Note that at each boosting iteration, instead of fitting all N samples, we can select only N' samples for training a SVR model, as suggested in [90], which can substantially reduce the computational complexity of each iteration as $N' \ll N$.

3.3.2 Boosting based Transfer Regression

Algorithm 2: BTMKR: Boosting based Transfer Multiple Kernel Regression

```
Input: Data set $\mathcal{S}_T$ from the target domain, number of iterations $T$, regularization
parameter $\lambda$, multiple kernel functions $\{F_1, \ldots, F_S\}$ learned from $S$ source domains, where
each $F_s$ is given by Eq. 3.6.
 1: Initialize residual: $r_n^1 = y_n, \forall n \in \{1, \ldots, N\}$, and $F = 0$
 2: for $t = 1, ..., T$ do
 3:     Compute the regression model $f^*$ and $h^*$ (line $8 - 21$)
 4:     Select the base learner: $f^t = \begin{cases} f^*, & \text{if } \frac{\sum_{n=1}^N r_n^t f^*(x_n)}{\lambda} > \sum_{n=1}^N r_n^t h^*(x_n) \\ h^*, & \text{otherwise.} \end{cases}$
 5:     Add $f^t$ to the ensemble: $F \leftarrow F + \epsilon f^t$
 6:     Update residuals: $r_n^{t+1} = y_n - F(x_n),\ \forall n \in \{1, 2, ...N\}$
 7: end for
Output: the final multiple kernel function $F(x)$

K-BTMKR
 8: for $s = 1, ..., S$ do
 9:     for $m = 1, ..., M$ do
10:         Fit the current current residuals: $\gamma_{s,m}^t = \frac{\sum_{n=1}^N r_n^t h_m^s(x_n)}{\sum_{n=1}^N h_m^s(x_n)^2}$
11:         Compute the loss of $h_m^s$: $e_{s,m}^t = \frac{1}{2} \sum_{n=1}^N \left(\gamma_{s,m}^t h_m^s(x_n) - r_n^t\right)^2$
12:     end for
13: end for
14: Fit the residuals by training a kernel regressor: $f^* = \arg\min_{f \in \mathcal{F}} \frac{1}{2} \sum_{n=1}^N \left(f(x_n) - r_n^t\right)$
15: Return the regression models: $f^*$ and $h^* = \arg\min_{\{h_m^s\}} e_{s,m}^t$

M-BTMKR
16: for $s = 1, ..., S$ do
17:     Fit the current current residuals: $\gamma_s^t = \frac{\sum_{n=1}^N r_n^t F_s(x_n)}{\sum_{n=1}^N F_s(x_n)^2}$
18:     Compute the loss of $F_s$: $e_s^t = \frac{1}{2} \sum_{n=1}^N \left(\gamma_s^t F_s(x_n) - r_n^t\right)^2$
19: end for
20: Fit the residuals by training a kernel regressor: $f^* = \arg\min_{f \in \mathcal{F}} \frac{1}{2} \sum_{n=1}^N \left(f(x_n) - r_n^t\right)$
21: Return the regression models: $f^*$ and $h^* = \arg\min_{\{F_s\}} e_s^t$
```

As explained in Section 1, we typically have very little data in the target domain, and therefore the model can easily overfit, especially if we train a complicated MKL model, even with the boosting approach. To deal with this issue, we can implicitly regularize the candidate functions at each boosting iteration by constraining the learning process within the function space spanned by the kernel functions trained on the source domains, rather than training the model in the function space spanned by arbitrary kernels. On the other hand, however, the underlying assumption of this approach is that at least one source domain is closely related to the target domain and therefore the kernel functions learned from the source domains can be reused. If this assumption does not hold, negative transfer could hurt the prediction performance. To avoid this situation, we also keep a MKL model which is trained only on the target domain. Consequently, the challenge becomes how to balance the knowledge embedded in the model learned from the source domains and the data fitting in the target domain.

To address this issue in a principled manner, we follow the idea of ϵ-boosting [101, 97] and propose the BTMKR algorithm, which is aimed towards transfer learning. There are two levels of transferring the knowledge of models: kernel-level transfer and model-level transfer, denoted by K-BTMKR and M-BTMKR respectively. At each iteration, K-BTMKR selects a single kernel function from $S \times M$ candidate kernels, while M-BTMKR selects a multiple kernel model from S domains. Therefore, K-BTMKR has higher "resolution" and more flexibility, at the price of higher risk of overfitting, as the dimension of its search space is M higher than that of M-BTMKR.

Kernel-Level Transfer (K-BTMKR)

Let $\mathcal{H} = \{h_1^1, \dots, h_M^1, \dots, h_1^S, \dots, h_M^S\}$ be the set of $S \times M$ candidate kernel functions learned from S source domains, and $\mathcal{F} = \{f_1, \dots, f_J\}$ be the set of J candidate kernel functions from the target domain. Note that as the kernel functions from the source domains are fixed, the size of $\mathcal{H}$ is finite, while the size of the function space of the target domain is infinite, since the weights learned by SVR can be arbitrary (i.e., Eq. 3.7). For simplicity of analysis, we assume J is also finite. Given the mapping $h : \mathbb{R}^d \rightarrow \mathbb{R}^{MS}, h(x) = [h_1^1(x), \dots, h_M^S(x)]^\top$ defined by $\mathcal{H}$ and the mapping f defined by $\mathcal{F}$, we formulate the transfer learning problem as:

$$\min_{\beta_\mathcal{S}, \beta_\mathcal{T}} \mathcal{L}\left(\beta_\mathcal{S}, \beta_\mathcal{T}\right) \quad \text{s.t.} \;\; ||\beta_\mathcal{S}||_1 + \lambda ||\beta_\mathcal{T}||_1 \leq \mu, \tag{3.9}$$

where $\mathcal{L}(\beta_{\mathcal{S}}, \beta_{\mathcal{T}}) \triangleq \sum_{n=1}^{N} \ell(\beta_{\mathcal{S}}^{\top} h(x_n) + \beta_{\mathcal{T}}^{\top} f(x_n), y_n)$, $\beta_{\mathcal{S}} \triangleq [\beta_1^1, \ldots, \beta_M^S]^{\top} \in \mathbb{R}^{MS}$, $\beta_{\mathcal{T}} \triangleq [\beta_1, \ldots, \beta_J]^{\top} \in \mathbb{R}^J$ are the coefficient vectors for the source domains and the target domain respectively, and λ is a parameter that controls how much we penalize $\beta_{\mathcal{T}}$ against $\beta_{\mathcal{S}}$. Intuitively, if the data from the target domain is limited, we should set $\lambda \geq 1$ to favor the model learned from the source domains, in order to avoid overfitting.

Following the idea of ϵ-boosting [102, 101], Eq. 3.9 can be solved by slowly increasing the value of μ by ϵ, from 0 to a desired value. More specifically, let $g(x) = [h(x)^{\top}, f(x)^{\top}]^{\top}$, and $\beta = \left[\Delta\beta_{\mathcal{S}}^{\top}, \Delta\beta_{\mathcal{T}}^{\top}\right]^{\top}$. At the t-th boosting iteration, the coefficient vector β is updated to $\beta + \Delta\beta$ by solving the following optimization problem:

$$\min_{\Delta\beta} \mathcal{L}\left(\beta + \Delta\beta\right) \quad \text{s.t.} \quad ||\Delta\beta_{\mathcal{S}}||_1 + \lambda||\Delta\beta_{\mathcal{T}}||_1 \leq \epsilon \tag{3.10}$$

As ϵ is very small, the objective function of Eq. 3.10 can be expanded to a first-order Taylor expansion, which gives

$$\mathcal{L}\left(\beta + \Delta\beta\right) \approx \mathcal{L}\left(\beta\right) + \nabla\mathcal{L}\left(\beta\right)^{\top} \Delta\beta, \tag{3.11}$$

where

$$\frac{\partial \mathcal{L}}{\partial \beta_j} = \sum_{n=1}^{N} -r_n^t g_j(x_n), \quad \forall j \in \{1, \ldots, MS + J\}. \tag{3.12}$$

Eq. 3.10 can be approximately solved [90]:

$$\Delta\beta_j = \begin{cases} \epsilon, & \text{if } j = \arg\max_j \frac{\sum_{n=1}^{N} r_n^t g_j(x_n)}{\lambda_j} \\ 0, & \text{otherwise} \end{cases}, \tag{3.13}$$

where $\lambda_j = 1, \forall j \in \{1, \ldots, S \times M\}$, and $\lambda_j = \lambda$, otherwise. In practice, as the size of function space of target domain is infinite, the candidate functions are actually computed by fitting the current residuals, as shown in Algorithm 2.

Model-Level Transfer (M-BTMKR)

The derivation of M-BTMKR is similar to that of K-BTMKR. Now we have $\mathcal{Z} = \{F_1, \ldots, F_S\}$ which is the set for S multiple kernel functions learned from S source domains. For M-

BTMKR, multiple kernel functions in $\mathcal{Z}$ serve as S candidate models for each boosting iteration.

$$\min_{\beta_{\mathcal{S}},\beta_{\mathcal{T}}} \mathcal{L}\left(\beta_{\mathcal{S}},\beta_{\mathcal{T}}\right) \quad \text{s.t.} \;\; ||\beta_{\mathcal{S}}||_1 + \lambda||\beta_{\mathcal{T}}||_1 \leq \mu, \tag{3.14}$$

The transfer learning problem is now defined in Eq. 3.14 which is the same as Eq. 3.9 except for a different definition for $\beta_{\mathcal{S}}$: $\beta_{\mathcal{S}} \triangleq [\beta_1, \ldots, \beta_S]^\top \in \mathbb{R}^S$, Now $\beta_{\mathcal{S}}$ is the coefficient vector for the multiple kernel functions learned from S sources.

3.3.3 Computational Complexity

The computational complexity of BMKR, as analyzed in [90], is $\mathcal{O}(TM\xi(N))$, where $\xi(N)$ is the computational complexity of training a single SVR with N samples. Standard learning approaches formulate SVR as a quadratic programming (QP) problem and therefore $\xi(N)$ is $\mathcal{O}(N^3)$. Lower complexity (e.g., about $\mathcal{O}(N^2)$) can be achieved by using other solvers (e.g., LIBSVM [106]). More important, BMKR can adopt a stochastic learning approach, as suggested in [90], which only selects N' samples for training a SVR at each boosting iteration. This approach yields a complexity of $\mathcal{O}(TM(N+\xi(N')))$, which makes the algorithm tractable for large-scale problems by choosing $N' \ll N$. The computational complexity of the BTMKR algorithms is $\mathcal{O}(TM(SN+\xi(N)))$. Note that in the context of transfer learning, we use all the samples from the target domain, as the size of the data set is usually small.

3.4 Experiments and Simulation Results

Electricity load consumption usually has complex nonlinear behaviors. Several factors may affect the house load profile. First, since human behaviors will have a significant effect on the load consumption, the load profiles for different day types will be quite different. For example, the load profile for residential houses for weekdays will be quite different from it on weekends. Meanwhile, the holiday and season effects may also affect the load profiles. Second, the weather conditions will affect the load profile, e.g., people will use more power for heating on colder days. As discussed in [107], both the temperature and wind speed would affect the electricity load consumption. Humidity and irradiance level may also be

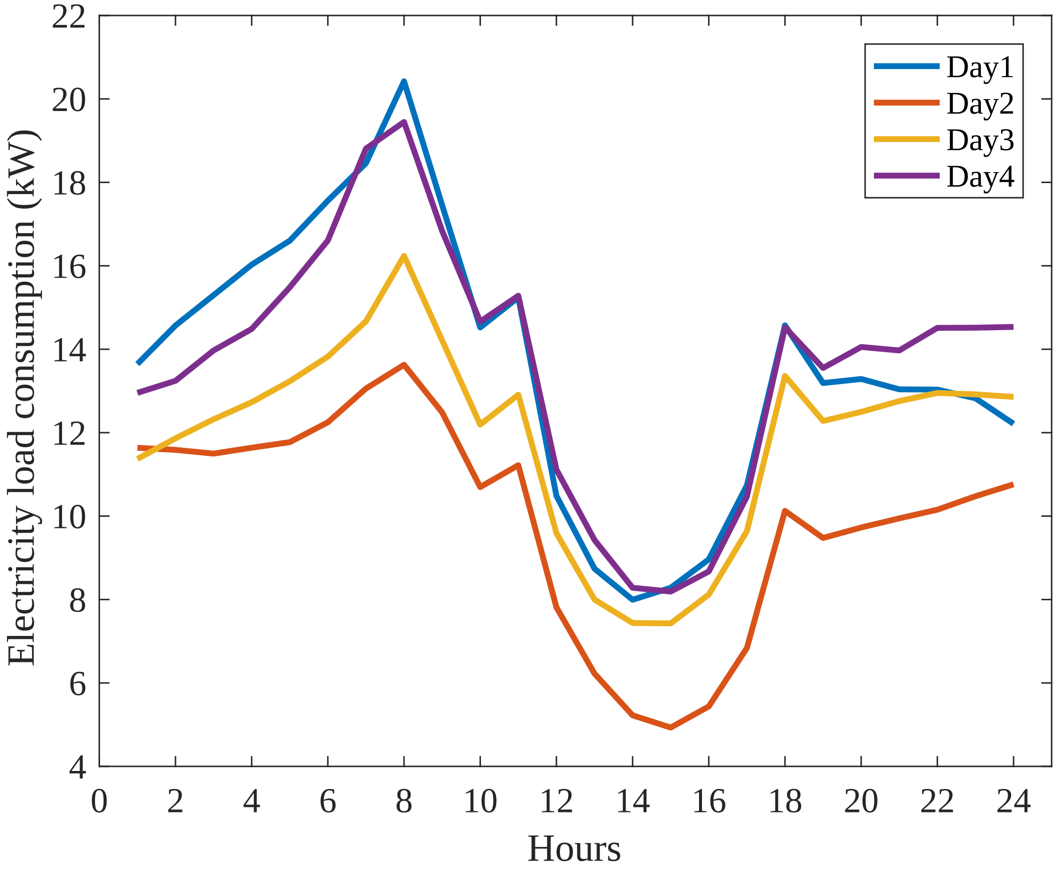

Fig. 3.2 Load consumption for four winter days

potential influencing factors. Third, the lagged load consumption itself is an important factor for short-term electricity load forecasting. Besides, the economic indicators may also be treated as an influencing factor for the load consumption.

In this section, we evaluate the proposed algorithms on the problem of short-term electricity load forecasting for residential houses. Several factors including day types, weather conditions, and the lagged load consumption itself may affect the load profile of a given house. In this chapter, we use three kinds of features for load forecasting: lagged load consumption, i.e., electricity consumed in the last three hours, temperature in the last three hours, and weekday/weekend information.

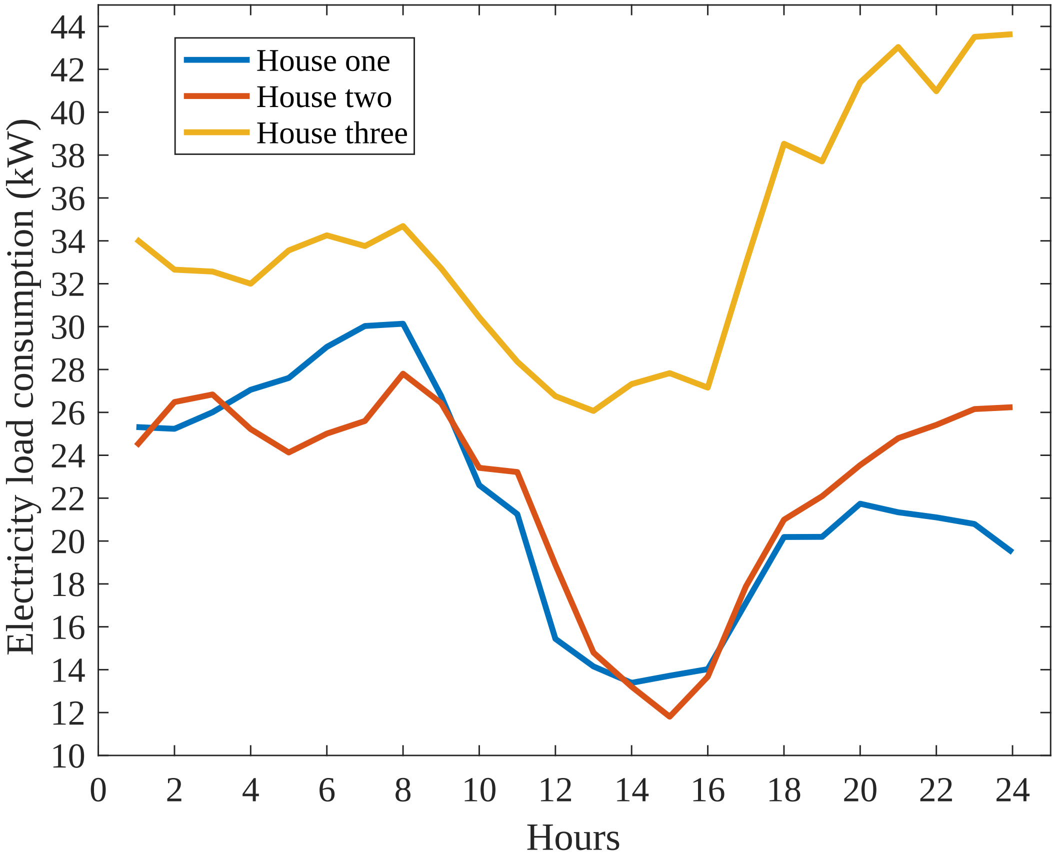

Fig. 3.3 Electricity load data for three houses

3.4.1 Data Description

The historical temperature data are obtained from [80], and the residential house load consumption data are provided by the US Energy department [108]. The data set includes hourly residential house load consumption data for 24 locations in New York state in 2012. For each location, it provides data for three types of houses, based on the house size: low, base, and high. Fig. 3.2 shows load consumption for a base type house for four consecutive winter days. We can see that the load consumption starts to decrease from 8 am and increases very quickly from 4 pm. Fig. 3.3 shows the load consumption for three high load consumption houses in nearby cities for the same winter day. It can be observed that the load consumption for house 1 is similar to house 2 and both are different from house 3.

Some pre-processing are implemented for the original data sets. Residential load files have been updated from 366 days in a year for leap years to the more general 365 days in a normal year. Then, in total, there are 8760 samples of hourly load consumption data for every house. Normalization are implemented for all the features with equation 3.15. In equation 3.15, x_i is the value for feature x at time i. x_{min} and x_{max} mean the maximal and minimal value for that feature. Mean Average Percentage Error (MAPE) shown in 3.16 is chosen as the comparison criteria. In this equation, y_i shows the real load consumption for time slot i and $\hat{y_i}$ is predicted load consumption for time slot i. NT means the total number of testing set.

$$x_i = \frac{x_i - x_{min}}{x_{max} - x_{min}} \tag{3.15}$$

$$MAPE = \frac{100}{NT} \sum_{i=1}^{NT} |\frac{y_i - \hat{y_i}}{y_i}| \tag{3.16}$$

3.4.2 BMKR for Electricity Load Forecasting

To test the performance of BMKR, we use the data of a high energy consumption house in New York City in 2012. We test the performance of BMKR separately for different seasons, and compare it with single kernel SVR and linear regression. We set the number of boosting iterations for the proposed algorithms to 100, the step-size of ϵ to 0.05, and the sampling ratio to 0.9. In order to accelerate the learning process, we initialize the model with linear regression. The candidate kernels for BMKR are: Gaussian kernels with 10 different widths $(2^{-4}, 2^{-3}, .., 2^{5})$ and a linear kernel. We repeat the simulation for 10 times, and each time we randomly choose 50% of the data in the season as training data and 50% of the data as testing data.

Table 3.1 shows the mean and standard deviation (std dev) of the Mean Average Percentage Error (MAPE) measurement for BMKR and the other two baselines. We can see that BMKR achieves the best forecasting performance for all seasons, obtaining 3.3% and 3.8 % average MAPE improvements over linear regression and single kernel SVR respectively.

Table 3.1 MAPE performance for high load consumption houses

Method	Spring	Summer	Fall	Winter	Average
Linear	10.42 ± 0.10	7.78 ± 0.13	9.21 ± 0.22	5.81 ± 0.13	8.30 ± 0.15
SVR	10.95 ± 0.21	7.73 ± 0.11	8.82 ± 0.21	5.88 ± 0.12	8.34 ± 0.16
BMKR	**10.31** ± 0.17	**7.64** ± 0.02	**8.42** ± 0.11	**5.73** ± 0.07	**8.02** ± 0.10

Table 3.2 Transfer learning MAPE performance for high load consumption houses

Method	Location 1	Location 2	Location 3	Location 4	Location 5	Location 6
Linear	8.02 ± 0.05	9.11 ± 0.70	17.39 ± 1.62	6.05 ± 0.02	11.43 ± 0.15	9.42 ± 0.65
SVR	11.53 ± 0.34	6.82 ± 0.39	25.90 ± 0.72	8.24 ± 0.08	26.31 ± 1.97	14.00 ± 0.65
BMKR	8.06 ± 0.03	6.64 ± 0.54	17.85 ± 1.31	5.29 ± 0.01	12.82 ± 0.21	9.05 ± 0.57
M-BTMKR	**5.35** ± 0.01	5.99 ± 0.02	**5.63** ± 0.19	**5.01** ± 0.01	9.13 ± 0.01	**5.69** ± 0.01
K-BTMKR	5.38 ± 0.02	**5.46** ± 0.30	6.97 ± 0.26	5.55 ± 0.09	**8.96** ± 0.14	7.31 ± 0.21

3.4.3 Transfer Regression for Electricity Load Forecasting

We evaluate the proposed transfer regression algorithms: M-BTMKR and K-BTMKR for three cases. In the first case, we randomly pick 6 high load consumption houses as target house and use the remaining 18 high load consumption houses as source houses. The second case is for base load consumption type houses and the third case is for low load consumption type houses, with the same arrangements as the first case. We repeat the simulation 10 times for each house, and each time we randomly choose 36 samples as the training data, and 100 samples as the testing data for the target house. For source houses, we randomly chose 600 data samples as the training data in each simulation. For K-BTMKR and M-BTMKR, λ is chosen by cross validation to balance the model leaned from source house data and the model learned from target house data.

The performance of the proposed two algorithms: M-BTMKR and K-BTMKR are compared with linear regression, single kernel SVR and BMKR. We use the same candidate kernels and boosting setting as described in Section 3.4.2. For the baselines, the forecasting models are trained only with data from target houses. Simulation results for three cases are shown in Table. 3.2, Table. 3.3, Table. 3.4. We can see that for all three cases, the proposed transfer algorithms could significantly improve the forecasting accuracy. For all houses in

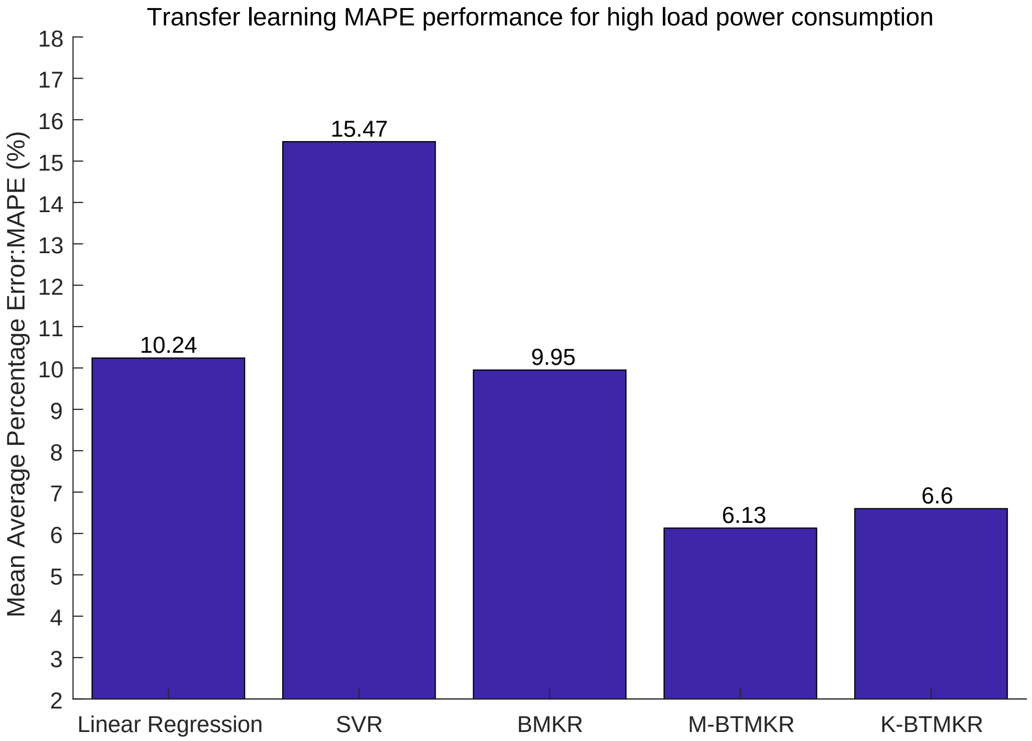

Fig. 3.4 Average transfer learning MAPE (%) performance for high load consumption houses

Table 3.3 MAPE performance on short-term load forecasting for base load consumption house

Method	Location 1	Location 2	Location 3	Location 4	Location 5	Location 6
Linear	8.43 ± 0.12	7.43 ± 0.19	15.77 ± 0.48	8.35 ± 0.12	10.47 ± 0.05	10.36 ± 0.11
SVR	10.11 ± 0.11	6.63 ± 0.13	16.27 ± 0.67	7.29 ± 0.04	12.73 ± 0.57	10.77 ± 0.13
BMKR	9.70 ± 0.43	8.03 ± 0.07	14.61 ± 0.49	5.78 ± 0.03	9.31 ± 0.10	9.90 ± 0.36
M-BTMKR	5.69 ± 0.03	**5.03** ± 0.02	**8.74** ± 0.02	7.36 ± 0.03	8.36 ± 0.02	5.58 ± 0.10
K-BTMKR	**5.63** ± 0.26	7.80 ± 0.04	8.76 ± 0.07	**5.57** ± 0.02	**7.71** ± 0.16	**5.42** ± 0.01

all three cases, the algorithm that shows best forecasting accuracy is either K-BTMKR or M-BTMKR. This shows that with the proposed transfer algorithms, we can successfully transfer the knowledge learned form source houses to target houses to improve the target house forecasting accuracy.

Figure 3.4, Figure 3.5, and Figure 3.6 show the average forecasting accuracy of different

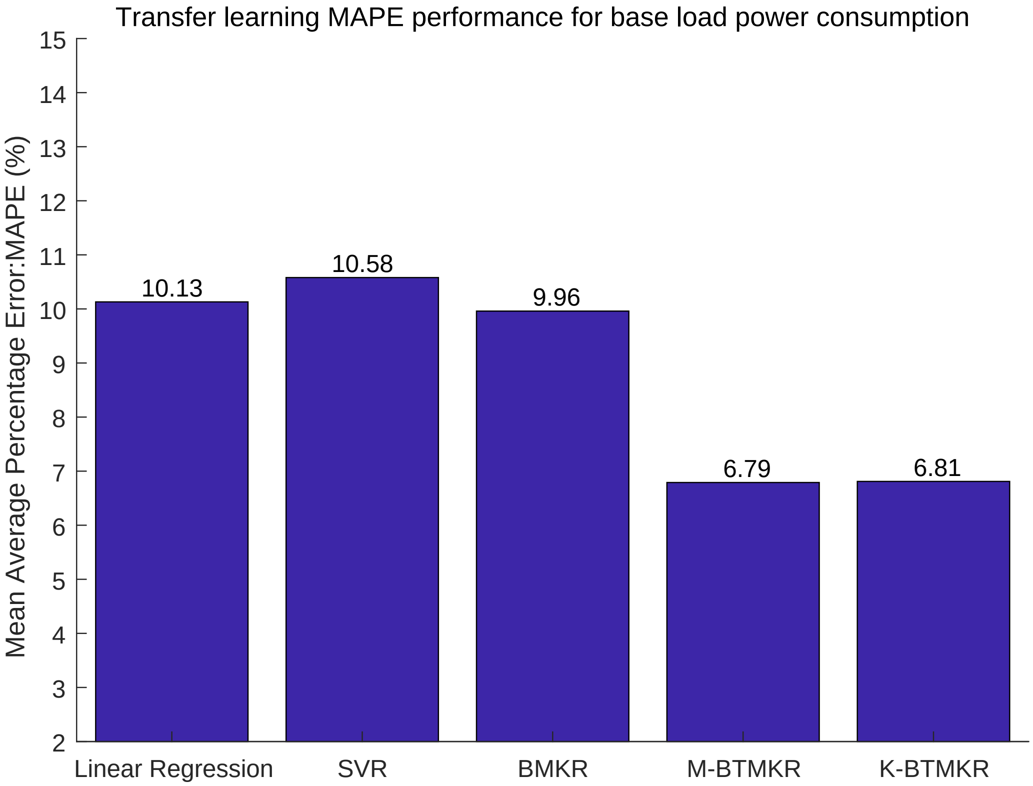

Fig. 3.5 Average transfer learning MAPE performance for base load consumption houses

Table 3.4 MAPE performance on short-term load forecasting for low load consumption houses

Method	Location 1	Location 2	Location 3	Location 4	Location 5	Location 6
Linear	9.33 ± 0.12	11.79± 0.66	15.40 ± 0.41	8.66 ± 0.07	14.43 ± 0.20	17.69 ± 1.73
SVR	10.33 ± 0.30	8.61 ± 0.23	13.75 ± 0.26	7.16 ± 0.15	18.11 ± 0.25	12.46 ± 0.44
BMKR	8.71 ± 0.16	12.61 ± 0.22	18.69 ± 0.25	8.87 ± 0.20	15.70 ± 0.35	13.66 ± 0.52
M-BTMKR	**6.65** ± 0.04	**7.89** ± 0.09	**6.59** ± 0.04	6.89 ± 0.12	**9.1** ± 0.17	7.38 ± 0.01
K-BTMKR	8.53 ± 0.05	8.61 ± 0.03	6.79 ± 0.03	**6.48** ± 0.08	10.06 ± 0.14	**7.12** ± 0.02

algorithms for three cases. M-BTMKR shows best performance in all three cases. The forecasting accuracy of M-BTMKR and K-BTMKR are very close and both are much better than linear regression, single kernel regression, and BMKR. On average, M-BTMKR shows 38.39%, 31.83%, 43.10% forecasting accuracy improvements over BMKR for case one,

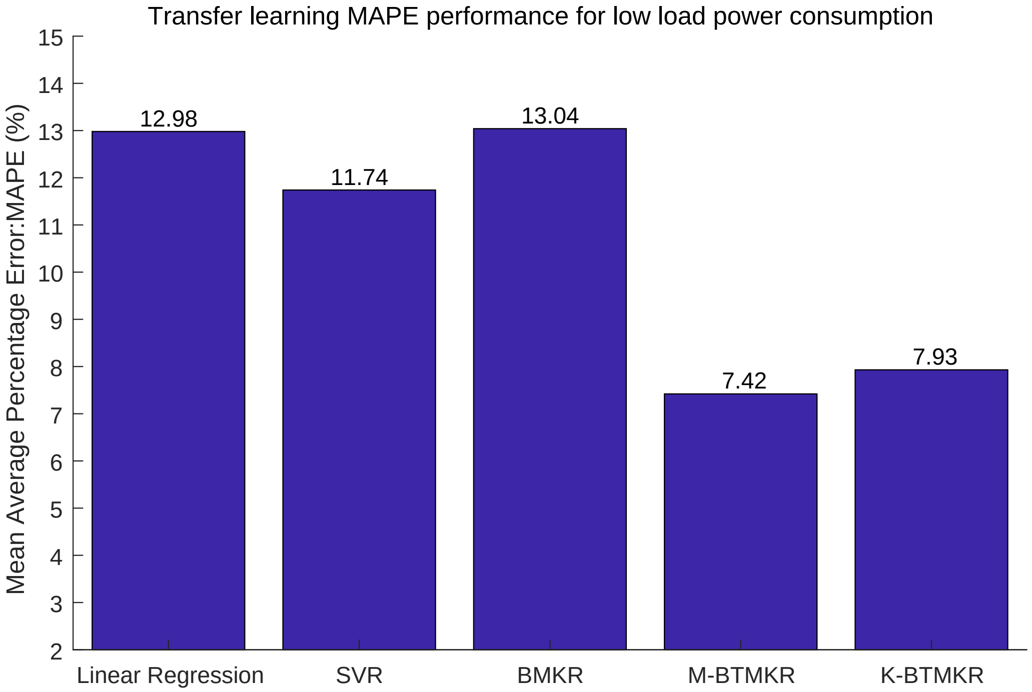

Fig. 3.6 Average transfer learning MAPE performance for low load consumption houses

two and three. This shows that we can get significant forecasting accuracy improvements for target houses with knowledge learned from source houses. As shown in Figure 3.7, either of the proposed transfer learning algorithm could significantly improve short-term electricity load forecasting performance.

Fig. 3.8 shows the average performance of single kernel SVR, linear regression and two proposed transfer algorithms for one high load consumption target house with different number of training samples from target house. We can see that, when the training data is very limited (10, 20), both the M-BTMKR and K-BTMKR can perform much better than single kernel SVR and linear regression. With the increasing number of training samples, the MAPE for single kernel SVR and linear regression are continuously reducing. When we have large amount of training samples (100), the performances of single kernel SVR model and linear regression model trained only with data from target house are very close to the performance of proposed transfer algorithms.

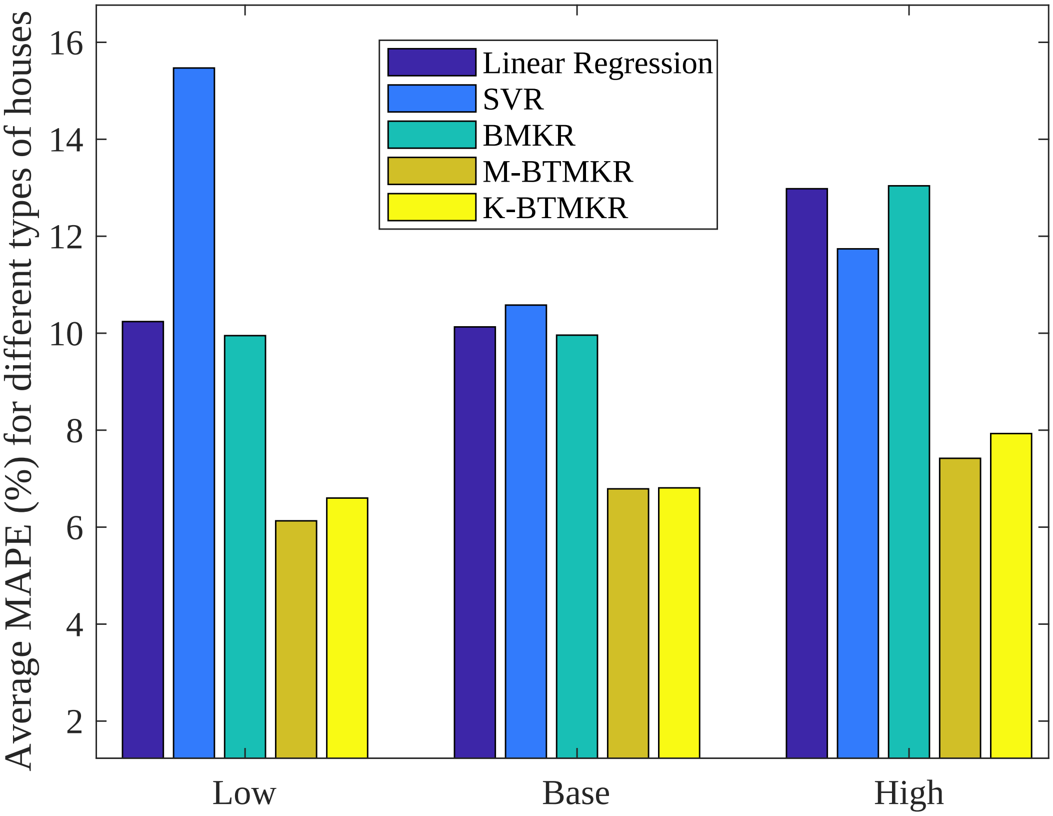

Fig. 3.7 Average MAPE for different algorithms for three cases

3.4.4 Negative Transfer Analysis

Sometimes the consumption pattern for source houses and target houses can be quite different. We would prefer that the transfer algorithms prevent potential negative transfer for such scenarios. Here we present a case study to show the importance of balancing the knowledge learned from source domains and data fitting in the target domain. We use the same high load target houses as described in Section 3.4.3, but for the source houses, we randomly chose eighteen houses from the low type houses. We repeat the simulation for 10 times and the results are shown in Table 3.5.

The proposed algorithms are compared with linear regression, single kernel SVR, BMKR, M-BTMKR$_{woT}$, and K-BTMKR$_{woT}$, where M-BTMKR$_{woT}$ and K-BTMKR$_{woT}$ denote the

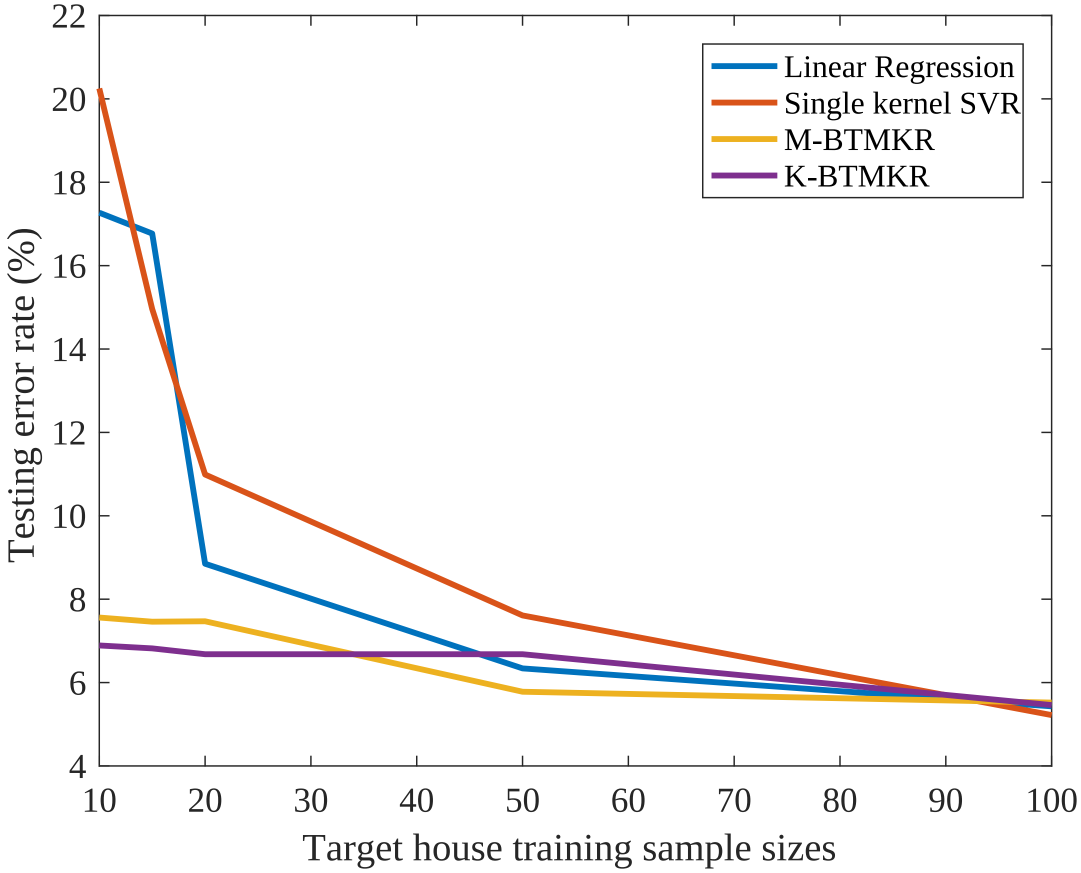

Fig. 3.8 Testing error with different training sample sizes.

BTMKR algorithms that we do not keep a MKL model trained on the target domain when we learn BTMKR models (i.e., we do not train f^* in Algorithm 2.). Simulation results show that, if we do not keep a MKL model trained on the target domain, we would encounter severe negative transfer problem, and the forecasting accuracy would be even much worse than the models learned without transfer. Meanwhile, we can see that the proposed M-BTMKR and K-BTMKR could successfully avoid such negative transfer. In this case, M-BTMKR and K-BTMKR still show better performance than other algorithms, though the forecasting accuracy of K-BTMKR is very close to BMKR. M-BTMKR achieves the best average forecasting performance and provides 14.37 % average forecasting accuracy improvements over BMKR. In summary, the BTMKR algorithms can avoid the negative

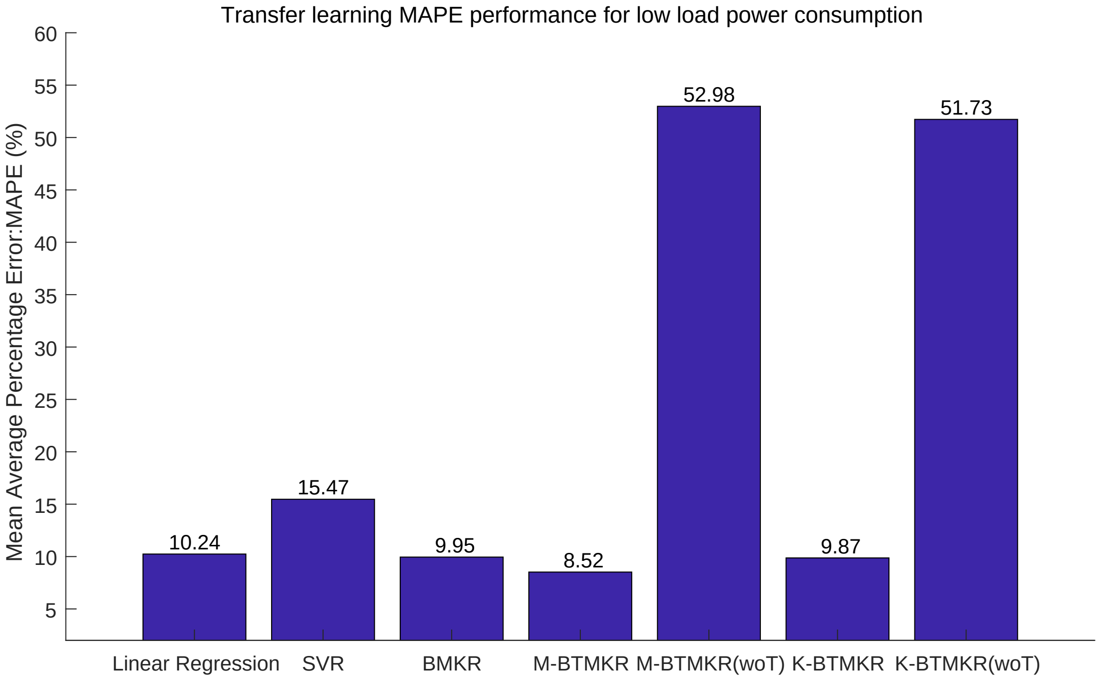

Fig. 3.9 Average transfer learning MAPE performance for high load consumption target houses with low load consumption source houses

transfer when the data distributions of source domain and target domain are quite different.

Table 3.5 Transfer learning MAPE performance for high load consumption target houses with low load consumption source houses

Method	Location 1	Location 2	Location 3	Location 4	Location 5	Location 6
Linear	8.02 ± 0.05	9.11 ± 0.70	17.39 ± 1.62	6.05 ± 0.02	11.43 ± 0.15	9.42 ± 0.65
SVR	11.53 ± 0.34	6.82 ± 0.39	25.90 ± 0.72	8.24 ± 0.08	26.31 ± 1.97	14.00 ± 0.65
BMKR	8.06 ± 0.03	**6.64** ± 0.54	17.85 ± 1.31	5.29 ± 0.01	12.82 ± 0.21	9.05 ± 0.57
M-BTMKR	**7.71** ± 0.01	8.74 ± 0.27	**8.65** ± 1.39	6.51 ± 0.52	**11.08** ± 0.21	**8.42** ± 0.86
M-BTMKR$_{woT}$	57.64 ± 0.05	59.02 ± 0.16	59.71 ± 0.53	46.25 ± 0.81	38.52 ± 0.02	56.71 ± 0.30
K-BTMKR	7.80 ± 0.06	8.60 ± 0.74	16.27 ± 2.48	**5.77** ± 0.16	11.42± 0.15	9.33 ± 0.67
K-BTMKR$_{woT}$	54.81 ± 0.05	58.31 ± 0.17	59.00 ± 0.11	43.95 ± 0.25	37.49 ± 0.12	56.81 ± 0.03

3.5 Chapter Summary

In this chapter, we first propose BMKR, a gradient boosting based multiple kernel learning framework for regression, which is suitable for short-term electricity load forecasting problems. Different from the traditional methods for MKL, the proposed BMKR algorithm learns the combination weights for each kernel using a boosting-style algorithm. The proposed BMKR algorithm avoids solving the complicated optimization problem of MKL. Simulation results on residential data show that the short-term electricity load forecasting could be improved with BMKR. We further extend the proposed boosting framework to the context of transfer learning and propose two boosting based transfer multiple kernel regression algorithms: K-BTMKR and M-BTMKR. Empirical results suggest that both algorithms can efficiently transfer the knowledge learned from source houses to the target houses and significantly improve the forecasting performance when the target houses and source houses have similar electricity load consumption patterns. We also investigate the effects of negative transfer and show that the proposed algorithms could prevent potential negative transfer when the source houses are different from the target houses.

In this chapter, it is shown that the proposed algorithms work well for short-term electricity load forecasting. In next chapter, we will show that model-free reinforcement learning algorithms can be used to deal with residential home energy management problem when only limited knowledge of system dynamics is available.

Chapter 4

Home Energy Management with Reinforcement Learning

4.1 Introduction

With the advancement of technology and increasing attention to environment protection, more and more electricity-driven products are introduced into people's daily life. Electric vehicles (EVs) are much more efficient than traditional internal combustion engine vehicles [9]. They are adopted by more and more consumers. However, the introduction of EVs brings a high power consumption burden on the power system and may even jeopardize the infrastructure of power grids if there is no proper energy management [7]. In the meantime, renewable energy generation is increasingly adopted for residential homes. Renewable energy generation such as wind and solar energy can provide home owener with cheap and clean energy, but it is quite intermittent and sensitive to weather conditions.

Smart grid using distributed renewable generation, advanced meters, energy storage, communication and computation tools can cope with these challenges. Energy management is a core issue for the smart grid [38] and can be beneficial for both the consumers and utility companies. Recently, with the development of home based energy storage and controllers, energy management for the residential sector has attracted more and more attention [29]. However, energy management for residential homes is a difficult problem. The main challenges come from uncertainties on both the power-supply and power-demand sides. Moreover, we usually only have very limited amount of load consumption data for residential houses which makes energy management an even more difficult task.

In this chapter, we consider a general case of home energy management with EV charging where only limited amount of historical data and battery specifications are available. We show how this problem can be formulated as a Markov Decision Process (MDP) that we propose to tackle using two model-free reinforcement learning algorithms: Neural Fitted Q Iteration (NFQ) and Deep Q-Network (DQN). The limited amount and heterogeneity of the available data makes implementing these algorithms challenging. We first show how a home simulator (**RLEnergy**) can be built from the available historical data and battery specifications, allowing us to enable the interactions with the RL algorithms needed for efficient exploration. The second challenge resides in how to incorporate the time series of base load power consumption in the state features of the MDP. Indeed, while only considering the power demand at the current time step may not be enough, incorporating too long a history into the state features could lead to overfitting [109].

To address this issue, we propose to model the base load power consumption using a recurrent neural network (RNN) and to enrich the state representation with both the RNN prediction for the next time step and its latent representation. The performance of the two algorithms are showcased on real-world data to optimize operating cost where they show significant improvements compared to previous rule based and batch RL methods.

Related work. In [110], the authors propose an approximate model for EV arrival and present a building energy management control algorithm with this model. In [111], the authors present a nonlinear model for a building cooling system and then build a control algorithm over this nonlinear model. The performance of these proposed control algorithms highly depends on the accuracy of system dynamics modeling which may not be realistic for residential homes where accurate system dynamics are usually unavailable. As EV batteries can be seen as distributed energy storage, EV charging scheduling has attracted significant attention, a survey on recent EV charging control strategies is presented [47]. In [55], the authors showed that with proper management, EV batteries can help stabilize the power grid and support large scale renewable energy adoption.

Reinforcement learning based control algorithms for smart grid are discussed in some recent papers. In [112], the authors propose a DQN based control strategy for storage devices in a microgrid. In [10], the authors propose to use Fitted Q-Iteration (FQI) to deal with smart home energy management. In [11, 113], the total power consumption of an EV charging fleet is learned by a batch reinforcement learning (BRL) algorithm and single EV charging is then scheduled with linear programming. To the best of our knowledge, there is no previous work studying the home energy management system (EMS) integrated with EV charging in one reinforcement learning framework.

Deep reinforcement learning (Deep RL) has strong state representation power and has shown successful applications on playing Atari, Go games and other complex control tasks [114, 115, 116]. With the development of distributed monitors and controllers, more and more data would be available for smart homes and the state space could be very complex. Deep RL algorithm, for its strong representation power, could be a promising candidate for home energy management where large amount of data are available. In this chapter, we aim to investigate the performance of Deep RL algorithm (DQN) and NFQ on home energy management integrated with EV charging.

Contributions and outline. In this chapter, we propose two RL based control algorithms to handle both the interactions with the power grid and EV charging scheduling in one unified RL framework. Our main **contributions** can be summarized as follows: 1) We propose an approach that can model smart home energy management with EV charging as a Markov decision process (MDP) [117]. 2) We tackle it with two model-free reinforcement learning algorithms: NFQ, DQN, and we investigate their performance on reducing operating cost and peak power consumption with real-world data.

After introducing relevant technical background in Section 4.2, the main components of smart home systems are introduced in Section 4.3. In Section 2.5, we show that energy management in smart homes can be formalized as an MDP and we propose two model-free RL based control algorithms to address it. Section 4.5 presents the experimental results with houses where only some historical data and battery specifications are available. Finally, the paper is concluded in Section 4.6.

4.2 Reinforcement Learning Background

4.2.1 Markov Decision Process

Home energy management can be seen as a sequential decision problem and can be modeled using a Markov Decision Process. Figure 4.1 [118] shows the agent environment interaction in an MDP. An MDP is a tuple $\langle S, A, T, R, \gamma \rangle$, where:

- S is a finite set of the states;
- A is a finite set of possible actions;
- $T : S \times A \times S \rightarrow [0, 1]$ is the transition probability from state s_t to state s_{t+1} when an action a_t is taken.
- $R : S \times A \rightarrow \mathbb{R}$ is the reward function, i.e. $R(s, a)$ is the reward received by the agent when taking action a in state s.
- γ is the discount factor $\gamma \in [0, 1]$

Solving an MDP means finding a policy $\pi : S \rightarrow A$, such that maximizing the expected return as shown in Equation 4.1.

$$G_t = \mathbb{E}[\sum_{j=0}^{\infty} \gamma^j r_{t+j}] \tag{4.1}$$

where r_t is the reward received at time step t.

4.2.2 Batch Reinforcement Learning

Reinforcement learning can be used to solve the MDP when little or no knowledge of the system dynamics is available. In batch reinforcement learning (BRL), the data collection and the learning process are decoupled and the control policy is learned from a set of learning experiences built from a set of historical data [119]. As BRL can reuse past experiences, it offers better data efficiency and converges faster than standard temporal difference methods. The main objective of BRL is to learn the best control policy given the existing batch of learning experiences.

Neural Fitted Q Iteration [120] is one of the most popular BRL algorithms. NFQ converts the learning from interactions paradigm to a series of supervised learning processes. There are mainly three phases for NFQ: exploration phase, training phase, and execution phase. In the exploration phase, a batch of transition samples $F = \{(s_t, a_t, r_t, s_{t+1}) | t = 1, \ldots, T\}$ are gathered. In the training phase, a training set D_{train} is built: it associates tuples (s, a) with estimated Q values $\overline{q}^h_{s,a}$. For each iteration h, the Q value $(\overline{q}^h_{s,a})$ for state action pair (s, a) is updated. A neural network is used to approximate the Q value function on D^h_{train}. In the execution phase, the policy learned in the training phase is applied.

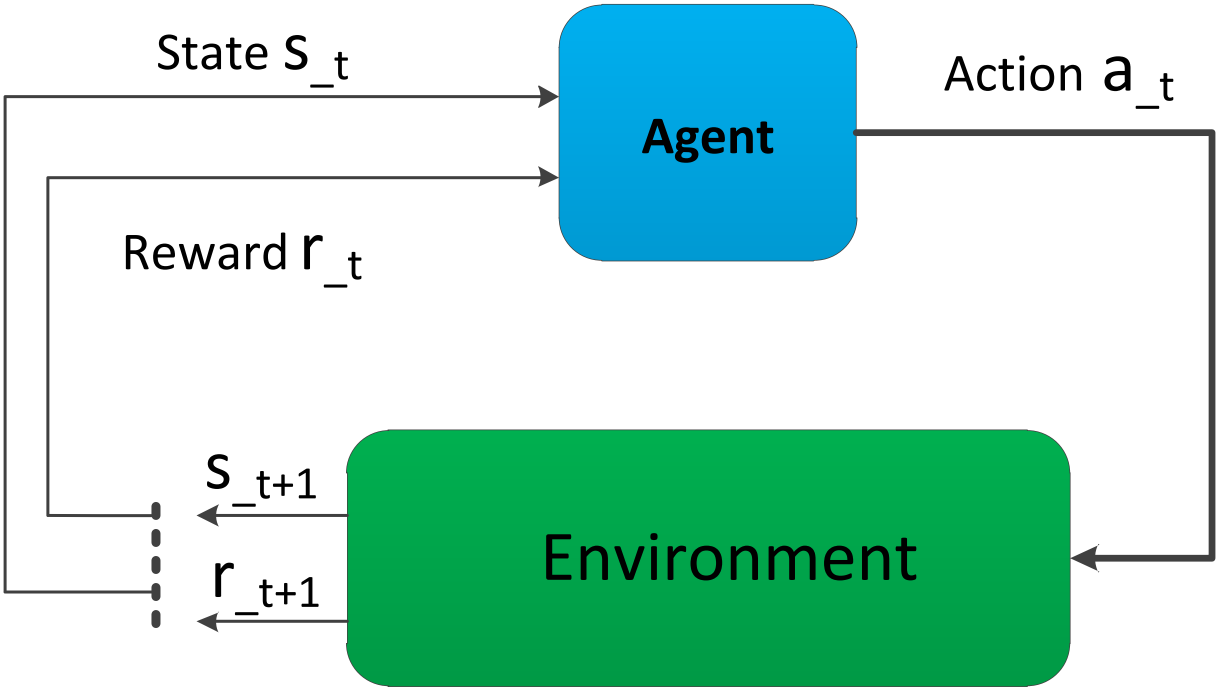

Fig. 4.1 Agent environment interaction in a Markov decision process

4.2.3 Deep Q Network

Reinforcement learning is known to be unstable when a nonlinear function such as neural networks is used as function approximator. There are several reasons for this: the sequence of observations for reinforcement learning are correlated and the data distribution will change during the learning process. This violates the assumption that the data should be independent and identically distributed. NFQ tackles instability issues by learning the function approximator using hundreds of iterations. However this method is inefficient for large neural networks.

Deep Q network (DQN) proposed in [114, 115], has successfully overcome the aforementioned challenges on combing deep learning with reinforcement learning framework. DQN uses a deep neural network to approximate the Q-value function and uses two techniques to tackle instability issues. It uses experience replay [121] and target network to reduce the correlations in the sequence observations and smooth the data distribution changes. It uses a deep neural network to approximate the Q value function $Q(s,a;\theta_i)$. θ_i are the parameters of the Q network. In the learning process, the Q-learning updates are based on a mini-batch of experience (s_t, a_t, r_t, s_{t+1}). This mini-batch is drawn uniformly at random from the pool of stored transition samples.

4.3 Smart Home Components

Figure 4.2 shows five main components of the smart home discussed in this chapter. These components are: home energy management system, renewable energy generation, power grid, home based battery storage, and power demand including base load power consumption and EV charging power consumption. Advanced meters are assumed to be installed in the home, enabling bi-directional communication.

4.3.1 Base Load Power Consumption

The power consumption of the smart home is divided into two groups: EV charging power consumption and base load power consumption. We consider the base load power consumption including all other power consumption in the home except the EV charging, namely power consumption for home appliances, cooling, heating, fans, interior lights, etc.

4.3.2 Electric Vehicle Charging Power Consumption

EV charging has become one of the major power demand for the residential sector with the fast increase of EV adoption. Here, we consider that each residential home is integrated with one EV and the EV is only charged at home. As shown in [122], EVs usually leave home around 7am and come back around 6pm. We apply the constraint that the EV needs to be charged with enough energy before 7am. Meanwhile, we assume that the EV can be charged with continuous rate which means any power rate ranging from zero to the maximal allowed charging rate. Figure 4.3 shows the total power consumption for one home with a Honda Fit EV [68]. We can see that without any management, EV charging would coincide with the peak hours of base load power consumption and further increase the peak of total power consumption. This effect could be even worse for some residential network, where early EV adopters may exhibit similar consumer behaviors in the communities and appear in clusters [8].

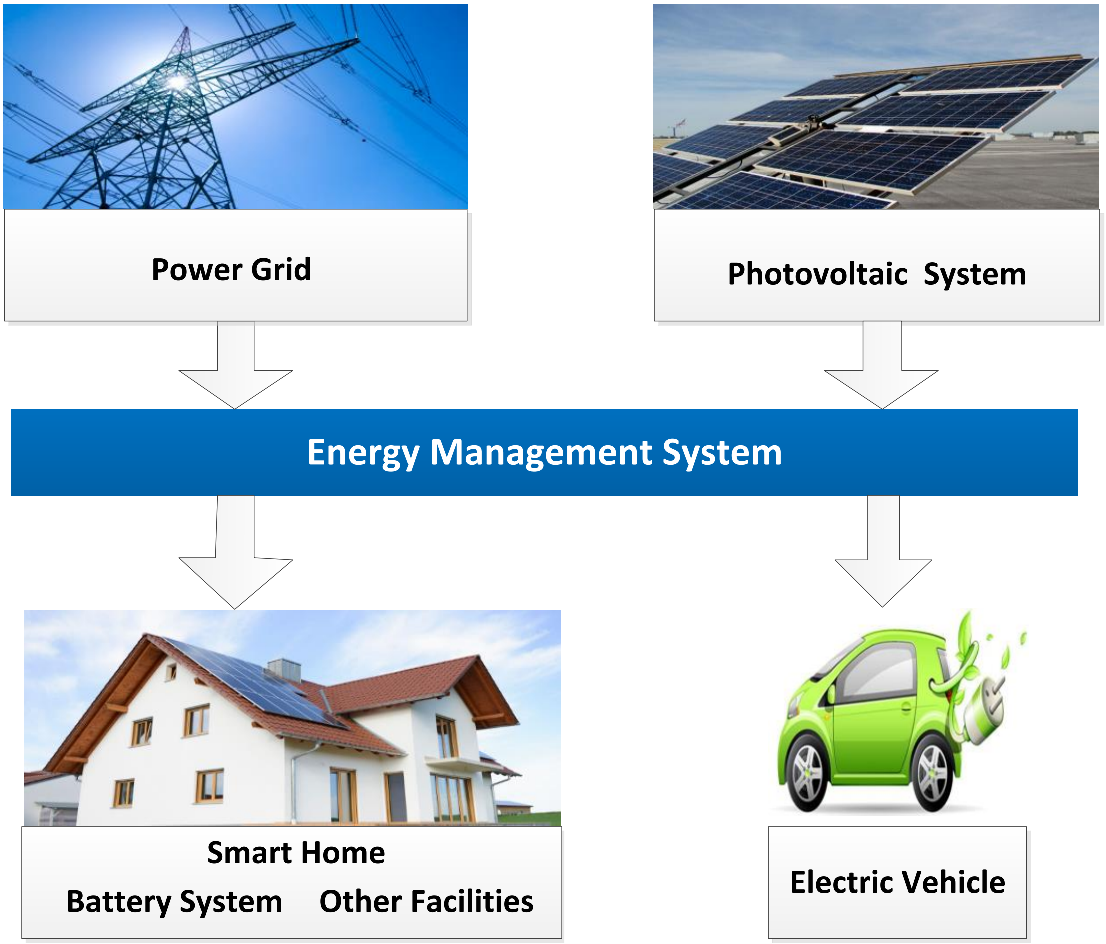

Fig. 4.2 Energy management system components for smart home

4.3.3 Renewable Energy Generation

Home based renewable energy generation including solar and wind power generation are becoming an important power source for more and more homes. In this chapter, we assume that the homes are equipped with solar panels which enable solar power generation. However, the output of solar power can be quite intermittent and varies according to different kinds of weather conditions.

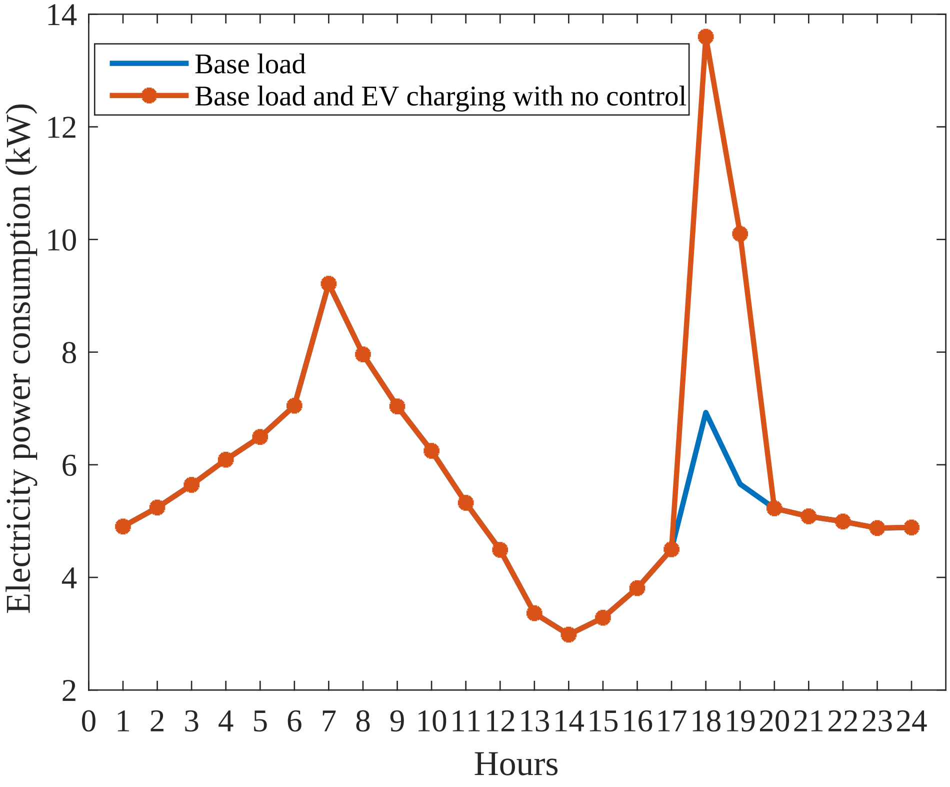

Fig. 4.3 Electricity demand with EV charging

4.3.4 Battery System

Home energy storage is an important component for smart homes. In this chapter, we assume that a home based battery system is installed. The battery system can be used to save the energy when there is power surplus for later use and mitigate the volatility of renewable energy generation.

4.4 Reinforcement Learning for Home Energy Management

In a home EMS, at every time step (each hour in this chapter) the decision maker has to interact with the power grid, the home battery charging scheduling and the EV charging scheduling. We start by showing how this sequential decision making problem can be formalized as an MDP and then propose two model-free reinforcement learning based control algorithms to solve it. In order to consider a general scenario we assume that we only have access to historical data of electricity demand, solar power generation, electricity price, EV arrival time, EV departure time and that battery specifications are available.

4.4.1 Energy Management as an MDP

States

Control actions for home EMS is determined by observing the current system state. Such a state (s_t) is composed of the following observations: home based battery state of charge (BSOC) ($H_{soc,t}$), solar power generation ($P_{v,t}$), electricity price (P_t), base load power consumption ($P_{base,t}$), home EV charging availability ($E_{a,t}$), time to departure for the EV ($E_{ttd,t}$), and current EV battery state of charge ($E_{soc,t}$). $H_{soc,t}$ is defined as 100% when the battery is fully charged and 0% when fully discharged. The last three state variables are EV related variable:. $E_{a,t}$ shows the charging availability for EV (set as 1 when EV is at home and 0 otherwise), $E_{soc,t}$ shows the battery state of charge of the EV battery, and $T_{ttd,t}$ shows how many hours are left before the departure of the EV. Then the MDP state at time step t, $s_t \in S$ is defined as: $s_t = (H_{soc,t}, P_{v,t}, P_t, P_{base,t}, E_{a,t}, E_{ttd,t}, E_{soc,t})$.

Actions

In this chapter, we implement control for the charging scheduling of home and EV batteries. We assume that both batteries can be charged or discharged at any continuous value of power from zero to the maximal allowed charging rate and that the home EMS can inject energy back into the power grid. For every time step, two control actions have to be taken $U_{buy,t}$ and $C_{ev,t}$. Positive values of $U_{buy,t}$ corresponds to energy bought from the grid and negative values to selling power back to the power grid. $C_{ev,t}$ corresponds to the EV charging rate, where negative values correspond to discharging the EV with $C_{ev,t}$. The power balance shown in Equation 4.2 must be satisfied for every time step (the left hand side is the power demand and the right hand side is the power supply). While $C_{ev,t}$ is the charging rate for the EV, $C_{hb,t}$ is the charging rate for the home battery and is decided deterministically as a function of $U_{buy,t}$ and $C_{ev,t}$.

$$C_{ev,t} + C_{hb,t} + P_{base,t} = P_{v,t} + U_{buy,t} \tag{4.2}$$

After these actions are taken, the state features corresponding to home and EV BSOC are updated using Equation 4.3 and 4.4 $H_{soc,t+1}$ and $E_{soc,t+1}$ shows the battery SOC for home battery and EV battery in time step $t+1$. $B_{hb,cap}$ is the battery capacity for home battery. $B_{ev,cap}$ is the EV battery capacity. L is the length for time slot. η_b and η_e are the energy conversion efficiency for home battery and EV battery. In this chapter, we assume the energy conversion efficiency is 0.9 for both two types of batteries.

$$H_{soc,t+1} = H_{soc,t} + \frac{C_{hb,t}}{B_{hb,cap}} \cdot L \cdot \eta_b \tag{4.3}$$

$$E_{soc,t+1} = E_{soc,t} + \frac{C_{ev,t}}{B_{ev,cap}} \cdot L \cdot \eta_e \tag{4.4}$$

At each time step, we first check whether the EV is in the home. If EV is not at home, all EV related variables are set to 0. If EV is at home, we need to determine the EV charging or discharging rate $C_{ev,t}$. In this chapter, we use four discretizations for both actions $U_{buy,t}$ and $C_{ev,t}$, leading to 16 possible actions at every time step.

Reward

The objective for home EMS is to reduce the long-term operating cost. We use negative cost as shown in Equation 4.5 as the MDP reward.

$$R_t = -Cost_t = -U_{buy,t} * P_t \tag{4.5}$$

While in practice, the buying and selling prices could be different for certain utility programs, we assume here that they are the same for the sake of simplicity.

4.4.2 Enriching the State Features with Recurrent Neural Networks

RL based control algorithms make control decisions based on current observations. It is intuitively clear that if future electricity demand is used, we could get better control policies. In this chapter, we use long short-term memory (LSTM) [123] recurrent neural network (RNN) to model short-term electricity load forecasting. After training an LSTM network with w hidden units to predict the future base load power demand from historical power consumptions, we can thus enrich the state features of the MDP at each time step t with the demand prediction. Moreover, we include the latent representation of the LSTM at this

Algorithm 3: NFQ based Home Energy Management System: NFQEMS

```
Require: Load $F = \{(s_t, a_t, r_t, s_{t+1}) | t = 1, ..., T\}$
Require: Define $\overline{Q}^0(s, a) = 0, \forall (s, a) \in F$, and $\overline{q}^h_{s,a} \in \overline{Q}^h(s, a)$
Require: Define $H$ as the Horizon to be performed
Require: Define $D^0_{train}$ as an initially empty training set
  h = 1;
  while $h \leq H$ do
    for all $(s_t, a_t, r_t, s_{t+1}) \in F$ do
      $\overline{q}^h_{s,a} = r_t + \gamma \max_{a \in A_{s(t+1)}} \overline{Q}^{h-1}(s_{t+1}, a)$;
      if $(((s_t, a_t)), ) \in D^{h-1}_{train}$ then
        $D^h_{train} \leftarrow D^{h-1}_{train} - \{((s_t, a_t), .)\}$;
      end if
      $D^h_{train} \leftarrow D^{h-1}_{train} \bigcup \{((s_t, a_t), \overline{q}^h_{s,a})\}$
    end for
    Implement supervised learning
    Use supervised learning to train a function approximator $\overline{Q}^h_{s,a}$ on the training set $D^h_{train}$
    $h \leftarrow h + 1$
  end while
  Use the learned policy for smart residential home energy management
```

time step which potentially encodes relevant information on the trend of the demand time series. For the LSTM network structure, we use one hidden layer with 10 hidden nodes. The MDP state with additional state features is now shown in Equation 4.6.

$$s(t) = (H_{soc,t}, P_{v,t}, P_t, P_{base,t}, E_{a,t}, E_{ttd,t}, E_{soc,t}, \hat{P}_{t+1}, \psi_L) \tag{4.6}$$

where $\hat{P}_{t+1}$ is the predicted based load power consumption for next time step and $\psi_L \in \mathbb{R}^w$ is the latent LSTM representation for time $t + 1$.

4.4.3 Neural Fitted Q Iteration based Home Energy Management

Neural fitted Q iteration (NFQ) uses a neural network to approximate the Q value function. NFQ allows the RL agent to learn a control policy from historical data. The NFQ based EMS control algorithm (NFQEMS) is defined in Algorithm 3.

To generate the training data, we first capture the historical data including solar energy generation, base load power consumption, EV arrival time, EV arrival BSOC, and EV departure time. We then assign a random state to the home battery for the first time step and take random actions from the allowed action sets for all following time steps, from which we get a set of transition experiences $F = \{(s_t, a_t, r_t, s_{t+1})\}$. In the training phase, NFQEMS learns an approximator for the Q value function using a training set $D_{train} = \{(s_t, a_t), q_t\}$ built from the transition set F in the following way. We first assign 0 to Q values for all state action pairs. We then learn an approximator with this initial training set D^0_{train}. In the following iterations of NFQEMS, we update the training set with updated Q values as shown in Equation 4.7. The training phase continues until either the maximum number of iterations H or a convergence criterion is reached. In this chapter, we use a feedforward neural network as the function approximator. We use a feedforward neural network with three hidden layers each with 128 neurons to approximate for the Q function. Rectifier linear units (ReLU) is used as activation functions. For NFQEMS, we set $H = 200$ and the convergence criteria is achieved when the variation of Q value is less than 5%: mean average percentage error for Q values of all state-action pairs of recent two iteration is less than 5%. In the execution phase, we can use a greedy policy with the learned Q value function approximator. The RL agent will choose an action from the allowed action set which has the highest Q value as follows, where γ is the discount factor.

$$\overline{q}^h_{s,a} = r_t + \gamma \max_{a \in A_{s(t+1)}} \overline{Q}^{h-1}(s_{t+1}, a) \tag{4.7}$$

4.4.4 Deep Q Networks based Home Energy Management

To use DQN for home energy management, we need a home simulator that the on-line RL algorithm can interact with. In this section, we first show how to build a smart home simulator, **RLEnergy**, from given historical data and battery specifications and then use this simulator to interact with DQN.

RLEnergy. Suppose we have a historical data set D_{his} for T time steps. As defined in Section 4.4.1, the state s_t is defined as $s_t = (H_{soc,t}, P_{v,t}, P_t, P_{base,t}, E_{a,t}, E_{ttd,t}, E_{soc,t})$. There are two kinds of state variables for s_t: fixed state variables and adaptive state variables. With given D_{his}, state variable $P_{v,t}$, P_t, $P_{base,t}$, $E_{a,t}$, $E_{ttd,t}$ are fixed for all time steps. $H_{soc,t}$, $E_{soc,t}$ will be updated by Equation 4.3 and Equation 4.4. Then we can build a simulator based on the historical data D_{his} and battery dynamics. For time step $t+1$, s_{t+1}, $p_{v,t+1}$, P_{t+1}, $P_{base,t+1}$, $E_{a,t+1}$, $E_{ttd,t+1}$ are taken from D_{his} directly according to time step index

Algorithm 4: DQN based Home Energy Management System: DQNEMS

```
Initialize replay memory $D = [empty\ set]$ as Capacity $N$
Initialize neural network $Q$ with random parameters $\theta$
Initialize target network $\hat{Q}$ with parameters $\theta^- = \theta$
for $episode = 1, K$ do
    Go to time step $t = 1$ with RLEnergy simulator and assign a random value for home battery
    for t = 1, T do
        With probability $\epsilon$ select a random action
        Otherwise select $a_t = \underset{a}{argmax} Q(s_t, a_t; \theta)$
        Execute action $a_t$ for home simulator and observe reward $r_t$ and next state $s_{t+1}$
        Store transition $(s_t, a_t, r_t, s_{t+1})$ in replay memory $D$
        Sample random minibatch $D_{mini}$ of transitions $(s_t, a_t, r_t, s_{t+1})$ from $D$
        Set $y_t = \begin{cases} r_t & if\ \ t = T - 1 \\ r_t + \gamma \underset{a_{t+1}}{max} \hat{Q}(s_{t+1}, a_{t+1}; \theta_i^-) & otherwise \end{cases}$
        Perform a gradient decent step on $(y_t - Q(s_t, a_t; \theta))^2$ with respect to network parameters $\theta$
        Update the target network parameters $\theta^-$ every $C$ steps: $\theta^- = \theta$
    end for
end for
Use the learned policy for smart residential home energy management
```

and $H_{soc,t+1}$ and $E_{soc,t+1}$ will be decided on the actions of $C_{hb,t}$ and $C_{ev,t}$ taken at time step t. This simulator will enable us to use on-line reinforcement learning algorithms such as DQN.

DQN based home energy control algorithm DQNEMS is defined in Algorithm 4. We first initialize a replay memory D with capacity N. For DQNEMS, we parameterize the Q-network with parameters θ and target neural network Q^- with θ^-. We first assign $\theta^- = \theta$. As shown in Algorithm 4, the outer loop learns DQN with K episodes, and the inner loop shows the parameter updating for every time step. T is the total number of time steps for the used historical data. Every episode will end when T is reached.

For every time step, RL agent will take a random action with probability ϵ for exploration or choose the action with maximal Q value estimate. After action a_t is taken, immediate reward r_t is received and the agent will go to next state s_{t+1}. Transition tuple (s_t, a_t, r_t, s_{t+1}) will then be stored to memory D. A mini-batch of transitions D_{mini} will be sampled randomly from D. Parameters for target network θ^{-} will be updated every C steps to stabilize the learning process. After K episodes, we can use the learned model for home energy management. To implement a fair comparison with NFQ, we use the same neural network structure as described in Section 4.4.3, except that the output layer has 16 neurons corresponding to the Q value for every possible action.

4.5 Experimental Results

4.5.1 Experiment Setup

In this section, we compare the performance of the proposed control algorithms with two baselines: rule-based and batch RL based RLbEMS which are described in [10]. The main differences between RLbEMS and our NFQEMS is that we consider the modeling and charging scheduling of EVs while RLbEMS only consider interactions between home and power grid.

Time-of-use (TOU) electricity price [78] and residential home power consumption for three houses in three locations in New York [70]are used. We have hourly electricity price and base load power power consumption for one year. Figure 4.4 shows load consumption on a winter day for one house and electricity price structure. We can see that there are two peak power consumption periods: 6-9am and 6-10pm. For TOU electricity price structure, price is different for different hour of the day. We can see that during the peak hours, electricity price is quite higher than that in the off-peak hours. We consider that there are 10 PV panels (peak power generation is 220W for each panel) for every house and use the irradiance data from [72] to generate the solar power generation for one year. In this chapter, we assume that every house has one EV (Honda Fit) [68] and one home based battery. The specifications for EV and home battery are shown in Table 4.1. EV data described in [122] is used to build EV arrival data for one year. We assume that both the home battery and EV battery can be charged and discharged with continuous values from zero to maximal allowed charging rate. We use the first 11 months as the training set while the remaining one month data is used as the testing set.

Table 4.1 Specifications for home battery and EV

Method	Capacity (kWh)	Maximal charging rate (kW)
Home Battery	10	1
EV	20	6.6

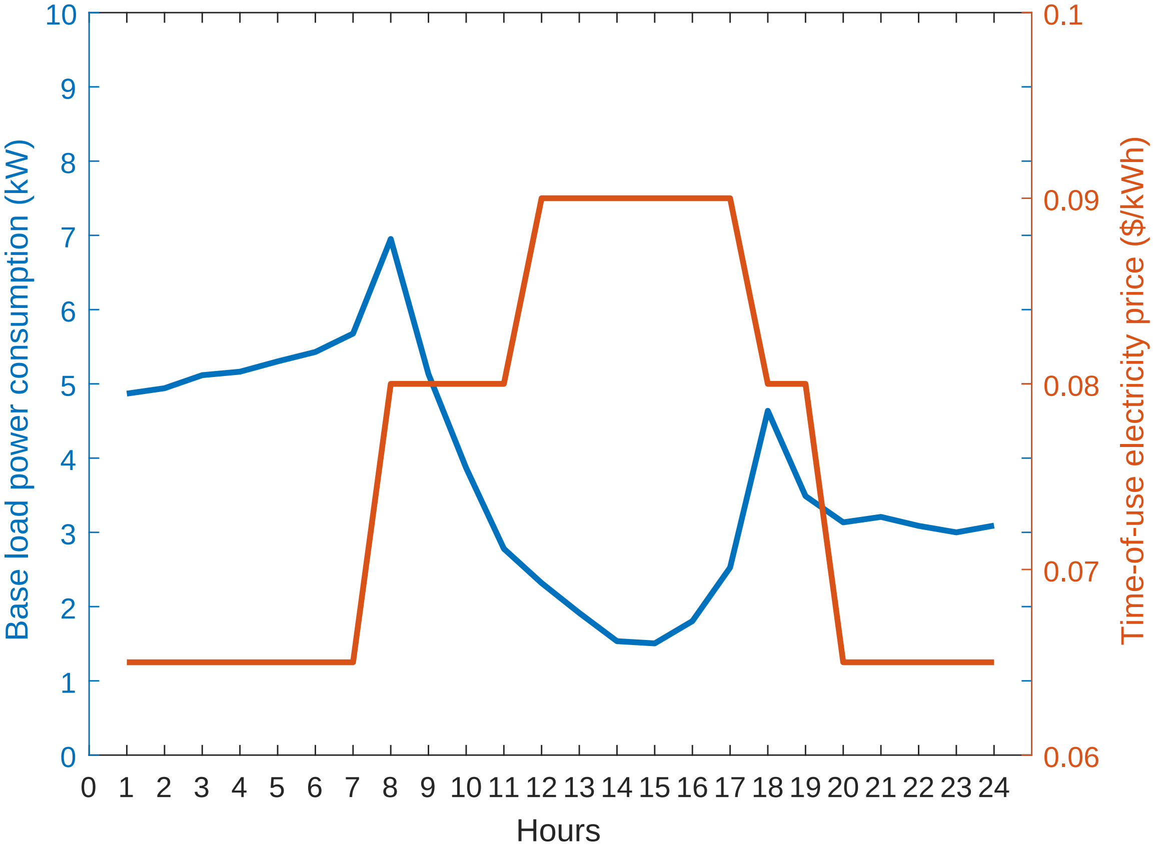

Fig. 4.4 Base load power consumption and time-of-use electricity price

4.5.2 Operating Cost and Peak Power Reduction

The total operating cost for the test set for three houses under different control algorithms are shown in Table 4.2. We can see that the two proposed RL based control algorithms can help reduce operating cost compared with two baselines. For NFQEMS it can reduce by 6.71% of the operating cost over rule-based control and 3.93% over RLbEMS control. For

DQNEMS it can reduce by 6.91% of the operating cost over rule-based control and 4.12% over RLbEMS control.

Table 4.3 shows that both NFQEMS and DQNEMS can help reduce average of daily operating cost for all houses. However, the standard deviation of average daily operating cost is quite high. This is probably due to the fact that the base load power consumption varies from day to day.

Table 4.2 Total operating cost for different control algorithms ($USD)

Method	Location 1	Location 2	Location 3	Average
Rule based	401.76	402.46	216.07	340.10
RLbEMS	389.36	392.41	208.95	330.24
NFQEMS	359.29	**385.64**	206.15	317.27
DQNEMS	**358.36**	386.26	**205.22**	**316.61**

Table 4.3 Daily average operating cost for different control algorithms ($USD)

Method	Location One		Location Two		Location Three	
	Mean	St.dev	Mean	St.dev	Mean	St.dev
Rule based	12.96	3.52	12.99	3.52	6.97	2.36
RLbEMS	12.56	3.41	12.66	3.42	6.74	2.29
NFQEMS	11.59	2.67	**12.44**	3.41	6.65	2.19
DQNEMS	**11.56**	2.67	12.46	3.42	**6.62**	2.29

Table 4.4 Daily average peak power consumption for different control algorithms (kW)

Method	Location One		Location Two		Location Three		Average	
	Mean	St.dev	Mean	St.dev	Mean	St.dev	Mean	St.dev
Rule based	16.42	1.90	17.00	2.55	13.08	1.72	15.50	2.06
RLbEMS	15.30	1.69	16.32	2.38	12.29	1.51	14.77	1.89
NFQEMS	**11.02**	1.90	11.63	2.55	**8.78**	1.74	**10.48**	1.74
DQNEMS	11.12	1.92	**11.41**	2.73	9.35	1.78	10.63	1.79

Peak power management is crucial for smart grid stability. Table 4.4 shows the daily peak power consumption for all three houses with different control algorithms. We can see

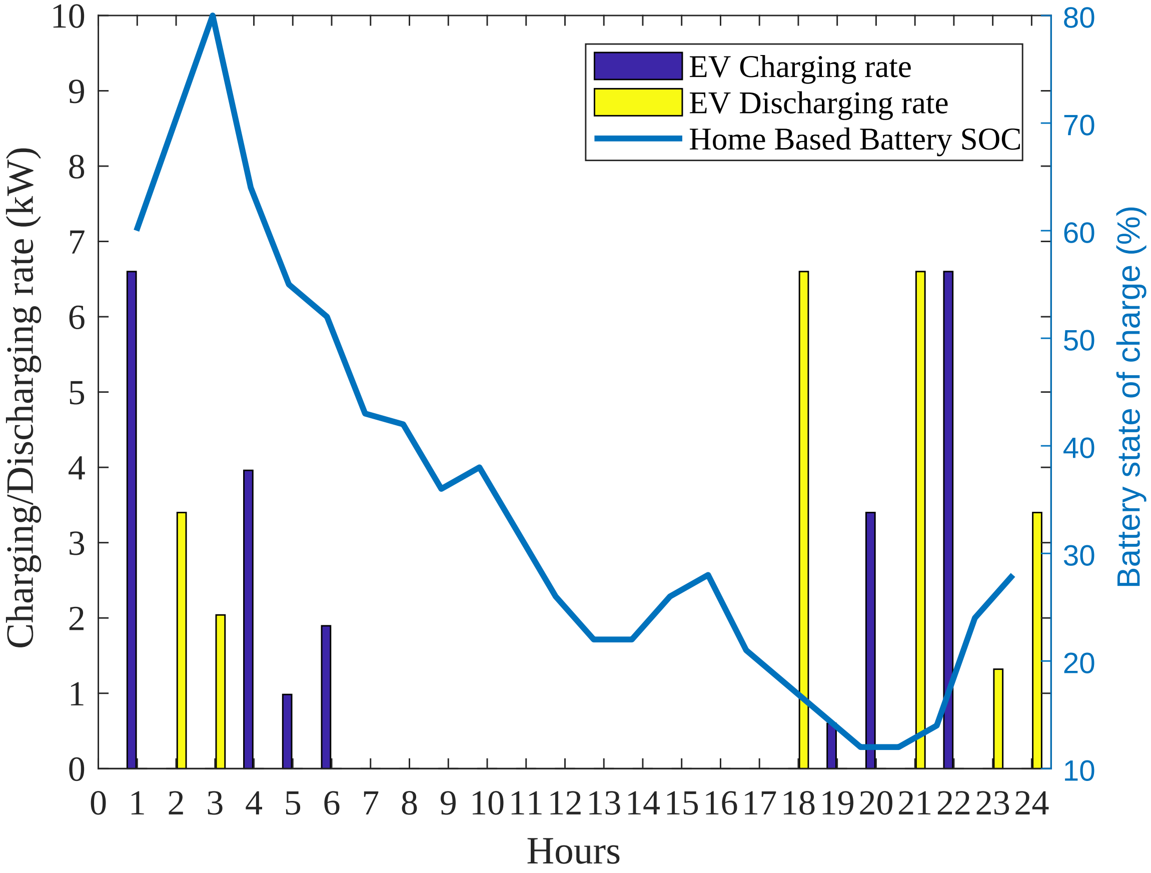

Fig. 4.5 Home based battery and EV battery energy state (for location 1)

that with proposed control algorithms, daily peak power consumption can be significantly reduced. For NFQEMS it can reduce 32.39% of daily peak power consumption over rule-based control and 29.05% over RLbEMS control. For DQNEMS it can reduce daily peak power consumption by 31.42% over rule-based control and by 27.69% over RLbEMS control. In the smart home MDP formalization, peak power is not part of the reward function. Meanwhile, electricity price structure will influence the immediate reward. This means that NFQEMS and DQNEMS can successfully learn the TOU price structure. It also suggests that TOU price structure will encourage cost-sensitive consumers to shift their deferable load to off-peak hours.

Figure 4.5 shows the EV charging, discharging and home battery energy state for house

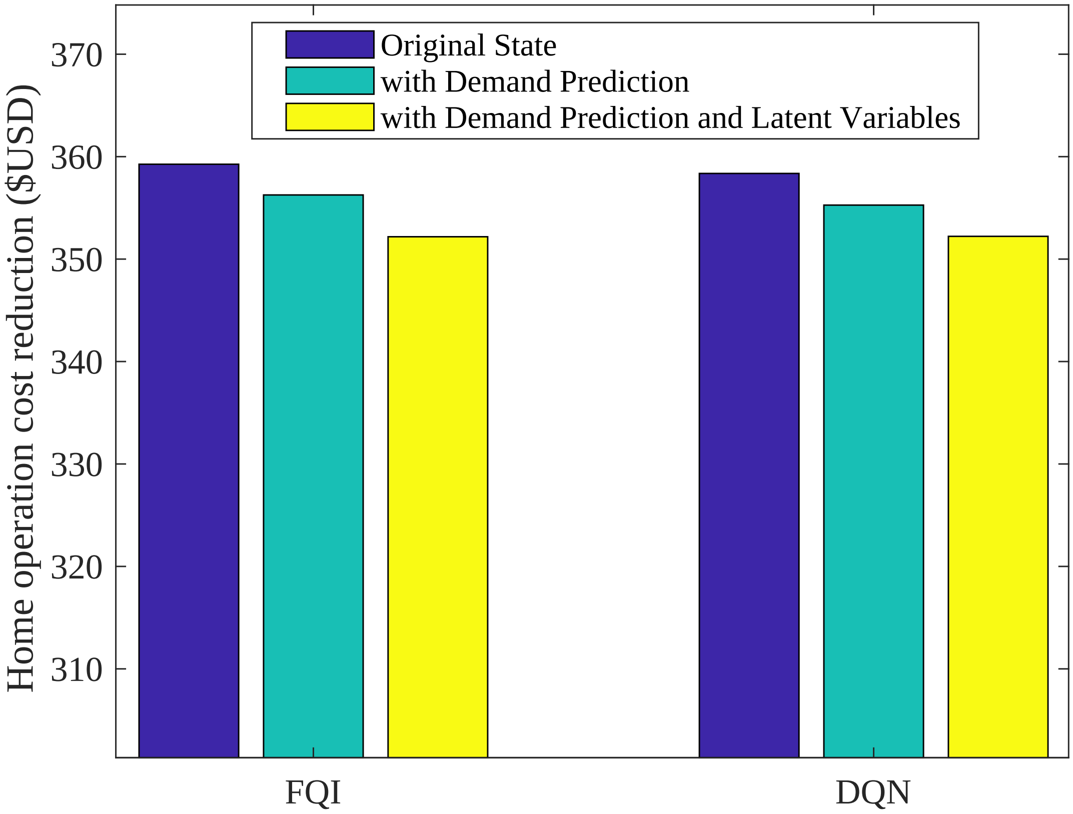

Fig. 4.6 Total operating cost with RNN predictions

in location 1 under DQNEMS control. We can see that EV charging is postponed from immediate charging. After EV arrival, home EMS will choose to discharge EV battery and then charge the energy back when electricity price is lower. This can help reduce the peak power consumption and operating cost. EV charging for NFQEMS is similar to Figure 4.5, EV battery discharging will be applied and charging will be postponed.

4.5.3 Enriching State Features with RNN Predictions

As presented in Section 4.4.2, we can use RNN predictions and latent representations to enrich the state features for RL algorithms. Figure 4.6 shows the operating cost for house in location 1 with original state features and with additional features of RNN predictions:

load predictions and latent representations under control of NFQEMS and DQNEMS. It can be observed that when more related useful information are added to the input, we could further reduce the total operating cost.

4.6 Chapter Summary

With recent progress in advanced meters development and the prevalence of distributed energy generation, more attention has been paid to smart residential home energy management. Meanwhile, the increasing adoption of EVs brings new challenges and opportunities for smart grid energy management. In this chapter, we show that the home energy management with EV charging can be formulated as an MDP. We propose two reinforcement learning based control algorithms: NFQEMS and DQNEMS to address it and show that home energy management can be dealt with both batch RL (off-line RL algorithm) and DQN (on-line RL algorithm). To use on-line RL algorithms, we need a simulator to interact with. We describe how to build a simulator with given historical data and battery specifications. Experiments based on real-world data show that the proposed two methods could significantly help in reducing operating cost (from 3.93% to 6.91%) as well as peak power consumption (from 27.69% to 32.39%) over two baselines. We further use the predictions of RNN and show that both the demand prediction and latent representations of RNN could help improve the performance of the proposed control algorithms. In next chapter, we will present a conclusion to summarize the major work of this thesis.

Chapter 5

Conclusion and Future Research

Fossil fuels based energy still accounts for the major part of total energy consumption: 85.0% of the world energy consumption, and 81.59% of the United States energy consumption in 2016. This has brought some serious environmental problems. Renewable energy generation and transportation electrification can help reduce the greenhouse gas emission and protect the environment. However, the increasing adoption of renewable energy and transportation electrification has brought serious challenges for the power systems. In this thesis, we aim to tackle challenges for energy management and short-term electricity load forecasting for office buildings and residential homes.

In this chapter, we summarize the main contributions, experimental results, and limitations of proposed approaches for each chapter. The potential future work of this thesis is also discussed.

5.1 Thesis Summary

In Chapter 2, we focus on tackling the challenges of office building energy management. We propose two energy management control strategies with stochastic programming framework and short-term electricity load forecasting for large office buildings. The proposed algorithms: SPLET and SAA_SPLET operate in two stages: day-ahead scheduling and real-time operation. Simulation results with real-world data demonstrate that the proposed control algorithms can provide significant reduction for building operation cost: 7.2% for SPLET and 6.9% for SAA_SPLET. It is shown that the annual total operation cost for the building could be further reduced when both vehicle batteries and stand-alone batteries are used. The contributions in Chapter 2 are summarized as follows:

- A new energy scheduling scheme is proposed to coordinate the integration of large office buildings with PV panels, large number of EVs, day-ahead power market and time-of-use electricity to optimize the energy management for the building.
- Two computationally efficient control algorithms, Stochastic Programming and Load forecasting for Energy management with Two stages (SPLET) and Sample Average Approximation based SPLET (SAA_SPLET) are proposed which can help reduce the operating cost of the office building. The uncertainties lie in both the demand-side and supply-side: the base load power consumption, renewable energy output, EV arrival time, EV departure time, and EV arrival state of charge (SOC) are addressed in the proposed algorithms. The effectiveness of the proposed algorithms are presented with real-world data.
- EV vehicle-to-building (V2B) and stand-alone battery system are considered as countermeasures for the mismatch between day-ahead power scheduling and actual real-time demand realization.

Accurate electricity load forecasting is of significant importance for the secure and economic operation of a power system. In Chapter 3, we look into short-term electricity load forecasting for residential homes. Gradient boosting framework is used to improve electricity load forecasting accuracy. For smart grid applications, we may only have limited amount of data available for our target house while we have a large amount of data available for some other houses which is suitable for use with transfer learning. Thus, we further extend the proposed Boosting based Multiple Kernel Regression (BMKR) to transfer learning context. Simulation results show that the proposed BMKR can improve one-step ahead electricity load forecasting accuracy over linear regression and SVR for 3.3% and 3.8% on Mean Average Percentage Error (MAPE). The simulation results also show that the proposed transfer learning based algorithms can significantly improve the forecasting performance and avoid potential negative transfer for homes when only limited data available at least of 31.83% MAPE over other baselines. For Chapter 3, we have the following contributions:

- We propose a boosting based learning framework to learn regression models with multiple kernels efficiently.

- We leverage the Multiple kernel learning (MKL) models learned from other domains with transfer learning to deal with electricity demand forecasting when only limited amount of data is available.
- Negative transfer between source domains and target domain is analyzed which shows that the proposed algorithm can prevent potential negative transfer.

Energy management is a core issue for smart grid and efficient energy management can be beneficial for both consumers and utility companies. Energy management for residential homes now is attracting more attention with the development of home based storage and controllers. In Chapter 4, we propose a modeling approach to model residential home energy management with EVs as a Markov Decision Process and investigate how model-free reinforcement learning algorithms can be used for residential home energy management. We also propose a method to build a home energy simulator using historical data and knowledge of battery specifications. Simulation results show that the proposed methods can help reduce both the operating cost (from 3.93% to 6.91%) and peak power consumption (from 27.69% to 32.39%) over two baselines. Contributions for Chapter 4 are summarized below:

- We propose to formulate the home energy management, including EV charging, as a Markov Decision Process.
- The proposed algorithms: NFQEMS (Neural Fitted Q based Energy Management Sysmtem) and DQNEMS (Deep Q Network based Energy Management System) are implemented for real-world residential home energy management problem.

The approaches proposed in this paper, using machine learning algorithms for electricity load forecasting and energy management, have shown their effectiveness on the given data-set. However, there are still several limitations for our methods which are described as follows. Our method proposed in Chapter 2 aims to tackle energy management for office buildings integrated with EV charging. Some uncertainties for building power consumption, EV behavior and renewable energy generation are considered. We assume that the EV arrival behavior could follow a certain distribution. This assumption could work for office buildings, since employees' behaviors are more regular. But for commercial building, people's behavior will be more complex and irregular.

In Chapter 3, we propose a Multiple Kernel Regression (MKR) based short-term electricity load forecasting algorithm and we further extend it to a transfer learning context. We show that we could improve short-term load forecasting with proposed kernel based K-BTMKR and model based M-BTMKR algorithms when we can use information learned from some similar source domains. The ground assumption for our transfer learning based algorithms is that: feature spaces for source domain and target domain are the same which makes homogeneous transfer learning available. However, for a more general case, feature space for source domains and target domain could be different. For example, for one house we may have more detailed power consumption data without temperature information and for the other one, we have aggregated level power consumption data and accurate temperature information. Then for this case, our proposed algorithms will not work.

In Chapter 4, we show that we can use model-free reinforcement learning algorithms to deal with home energy management with EV charging. In the proposed algorithms, we use discretized control actions for electric vehicle charging and interactions with power grid. The discretized control action setting makes our approaches have a limitation for dimensionality, notably the curse of dimensionality: the number of actions increases exponentially with the number of degrees of freedom. For example, in our setting, we have a 2-degree of freedom and four-discretized control actions system and in total our action space has $4^2 = 16$ dimensionality. If we have 8-degree freedom system, the dimensionality of action space will be: $4^8 = 65536$. This will be even worse if we need a more finer control of actions as they require a correspondingly finer grained discretization, leading to an explosion of the number of discrete actions.

Overall, the research presented in this paper is focused on high level control algorithm, more fine grained power constraints, for example: frequency control is not considered. The above described limitations show that there is still a lot of interesting research that could be done to further improve the efficiency and security of smart grid with machine learning.

5.2 Future Research

Some potential research directions for future work are summarized as follows:

Benefits of using transfer learning for smart grid should be further studied. In Chapter 3, we assume that the feature spaces for the source domain and target domain are the same. In some other scenarios, the feature space for source domain and target domain can be different. Then, heterogeneous transfer could help deal with this setting. Heterogeneous transfer learning is used for scenarios where the source and target domains are represented in different feature spaces.

Besides, we assume that domain distributions for target domain and source domain are the same. If the domain distributions are not equal, then further domain adaptation will be needed. Domain adaptation process attempts to alter the distribution of the source closer to the target.

In our work in Chapter 3, we aim to transfer the learned electricity load forecasting models from source domains to target domain. Meanwhile, to learn a control policy usually will require a lot of data, especially for reinforcement learning algorithms. It would be ideal, if we could successfully transfer reinforcement learning based control policies from related source domains to target domain. For example, we can transfer the energy management control policies from some existing building with a large amount of data available to newly built buildings where little data is available.

Our work, in Chapter 4, using discretized control actions, may encounter dimensionality issues for high order freedom system. Reinforcement learning with continuous control actions could help relieve the dimensionality curse discussed in Section 5.1. We can leverage deep reinforcement learning algorithms proposed in [116, 124] to further improve energy management in the smart grid.

Most of recent deep reinforcement learning successes are only in single agent domains while many real-world applications would involve interactions between different agents and require large-scale distributed control. Large-scale and complex decision-making problems are still very challenging for RL algorithms. Hierarchy Reinforcement Learning (HRL) could help to tackle these challenges. Smart grid is a complex electrical network. From a higher level, there are hierarchy structures for power systems and many power systems are interconnected. At a lower level, for a certain neighgorhood level network, there maybe many residential houses in this network. If we only consider a single home, we may realize good peak power reduction for one specific home but we may not be able to guarantee that the whole power consumption would be reasonable.

Then another interesting research direction would be using large-scale reinforcement learning based control algorithms for more complex smart grid problems. Several work [125, 126] has already been done in this field, but more in-depth studies are still needed to accelerate the learning process and improve the control performances.

References

[1] BP Global, Accessed February 3, 2018, "Bp statistical review of world energy 2017," https://www.bp.com/content/dam/bp/en/corporate/pdf/energy-economics/statistical-review-2017/bp-statistical-review-of-world-energy-2017-full-report.pdf.

[2] US Energy Information Administration, Accessed February 3, 2018, "Americans use many types of energy," https://www.eia.gov/energyexplained/?page=us_energy_home.

[3] Uited States Enviromental Protection Agency, Accessed February 3, 2018, "Global greenhouse gas emissions data," https://www.epa.gov/ghgemissions/global-greenhouse-gas-emissions-data.

[4] US Energy Information Administration, Accessed February 3, 2018, "How the united states uses energy," https://www.eia.gov/energyexplained/index.cfm?page=us_energy_use.

[5] European Commission, Accessed February 3, 2018, "European climate action," http://en.openei.org/wiki/Main_Page.

[6] International Energy Agency, Accessed February 3, 2018, "Ev outlook," https://www.iea.org/publications/freepublications/publication/GlobalEVOutlook2017.pdf,Accessed:January30,2018.

[7] D. Wu, H. Zeng, and B. Boulet, "Neighborhood level network aware electric vehicle charging management with mixed control strategy," in *Electric Vehicle Conference (IEVC), 2014 IEEE International*. IEEE, 2014, pp. 1–7.

[8] M. E. Kahn and R. K. Vaughn, "Green market geography: The spatial clustering of hybrid vehicles and leed registered buildings," *The BE Journal of Economic Analysis & Policy*, vol. 9, no. 2, 2009.

[9] R. C. Green, L. Wang, and M. Alam, "The impact of plug-in hybrid electric vehicles on distribution networks: A review and outlook," *Renewable and Sustainable Energy Reviews*, vol. 15, no. 1, pp. 544–553, 2011.

[10] H. Berlink and A. H. Costa, "Batch reinforcement learning for smart home energy management." in *IJCAI*, 2015, pp. 2561–2567.

[11] A. Chiş, J. Lundén, and V. Koivunen, "Reinforcement learning-based plug-in electric vehicle charging with forecasted price," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 5, pp. 3674–3684, 2017.

[12] M. Ozay, I. Esnaola, F. T. Y. Vural, S. R. Kulkarni, and H. V. Poor, "Machine learning methods for attack detection in the smart grid," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 8, pp. 1773–1786, 2016.

[13] C. Cecati, G. Mokryani, A. Piccolo, and P. Siano, "An overview on the smart grid concept," in *IECON 2010-36th Annual Conference on IEEE Industrial Electronics Society*. IEEE, 2010, pp. 3322–3327.

[14] US Energy Department, Accessed February 3, 2018, "Demand response," https://www.energy.gov/oe/activities/technology-development/grid-modernization-and-smart-grid/demand-response.

[15] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Springer series in statistics Springer, Berlin, 2001, vol. 1.

[16] N. M. Nasrabadi, "Pattern recognition and machine learning," *Journal of electronic imaging*, vol. 16, no. 4, p. 049901, 2007.

[17] R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, *Machine learning: An artificial intelligence approach*. Springer Science & Business Media, 2013.

[18] S. Jurado, À. Nebot, F. Mugica, and N. Avellana, “Hybrid methodologies for electricity load forecasting: Entropy-based feature selection with machine learning and soft computing techniques,” *Energy*, vol. 86, pp. 276–291, 2015.

[19] B. Yildiz, J. Bilbao, and A. Sproul, “A review and analysis of regression and machine learning models on commercial building electricity load forecasting,” *Renewable and Sustainable Energy Reviews*, vol. 73, pp. 1104–1122, 2017.

[20] D. Niu, Y. Wang, and D. D. Wu, “Power load forecasting using support vector machine and ant colony optimization,” *Expert Systems with Applications*, vol. 37, no. 3, pp. 2531–2539, 2010.

[21] E. C. Kara, M. Berges, B. Krogh, and S. Kar, “Using smart devices for system-level management and control in the smart grid: A reinforcement learning framework,” in *Smart Grid Communications (SmartGridComm), 2012 IEEE Third International Conference on.* IEEE, 2012, pp. 85–90.

[22] N. Shahid, S. Aleem, I. H. Naqvi, and N. Zaffar, “Support vector machine based fault detection & classification in smart grids,” in *Globecom Workshops (GC Wkshps), 2012 IEEE.* IEEE, 2012, pp. 1526–1531.

[23] H. Livani and C. Y. Evrenosoglu, “A machine learning and wavelet-based fault location method for hybrid transmission lines,” *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 51–59, 2014.

[24] J. C. Palomares-Salas, J. J. G. de la Rosa, A. Agüera-Pérez, and J. M. Sierra-Fernández, “Smart grids power quality analysis based in classification techniques and higher-order statistics: Proposal for photovoltaic systems,” in *Industrial Technology (ICIT), 2015 IEEE International Conference on.* IEEE, 2015, pp. 2955–2959.

[25] F. Uçar, Ö. F. Alçin, B. Dandil, and F. Ata, “Machine learning based power quality event classification using waveletˇŞentropy and basic statistical features,” in *Methods and Models in Automation and Robotics (MMAR), 2016 21st International Conference on.* IEEE, 2016, pp. 414–419.

[26] T. Chen, K. Qian, A. Mutanen, B. Schuller, P. Järventausta, and W. Su, "Classification of electricity customer groups towards individualized price scheme design," in *Power Symposium (NAPS), 2017 North American*. IEEE, 2017, pp. 1–4.

[27] D. Koolen, N. Sadat-Razavi, and W. Ketter, "Machine learning for identifying demand patterns of home energy management systems with dynamic electricity pricing," *Applied Sciences*, vol. 7, no. 11, p. 1160, 2017.

[28] US Energy Information Administration, Accessed February 3, 2018, "Energy consumed by building," https://www.eia.gov/tools/faqs/faq.php?id=86&t=1.

[29] A. T. Kaliappan, S. Sathiakumar, and N. Parameswaran, "Flexible power consumption management using q learning techniques in a smart home," in *Clean Energy and Technology (CEAT), 2013 IEEE Conference on*. IEEE, 2013, pp. 342–347.

[30] Ministerial, Clean Energy, Accessed February 3, 2018, "Ministerial, clean energy, global ev outlook understanding the electric vehicle landscape to 2020, clean energy ministerial. report 1 (2016)," https://www.iea.org/publications/freepublications/publication/Global_EV_Outlook_2016.pdf.

[31] . Erik Palm, Accessed February 3, *Study: Electric cars not as green as you think*, CNET, [Online] http://www.cnet.com/news/study-electric-cars-not-as-green-as-you-think/.

[32] California PEV Collaborative, Accessed February 3, 2018, "Communication guide 7," http://www.pevcollaborative.org/sites/all/themes/pev/files/Comm_guide7_122308.pdf.

[33] D. Santini, "Broad overview of plug-in hybrids and analytical studies," in *US department of energy plug-in hybrid vehicle discussion meeting, US Department of Energy, Washington (DC)*, 2006.

[34] US Department of Energy, Accessed February 3, 2018, "Workplace charging challenge progress update 2014: Employers take charge," http://www.afdc.energy.gov/uploads/publication/wpc_2014_progress_report.pdf.

[35] G. S. Pavlak, G. P. Henze, and V. J. Cushing, "Optimizing commercial building participation in energy and ancillary service markets," *Energy and Buildings*, vol. 81, pp. 115–126, 2014.

[36] F. Kamyab, M. Amini, S. Sheykhha, M. Hasanpour, and M. M. Jalali, "Demand response program in smart grid using supply function bidding mechanism," *IEEE Transactions on Smart Grid*, vol. 7, no. 3, pp. 1277–1284, 2016.

[37] K. Clement-Nyns, E. Haesen, and J. Driesen, "The impact of charging plug-in hybrid electric vehicles on a residential distribution grid," *IEEE Transactions on Power Systems*, vol. 25, no. 1, pp. 371–380, 2010.

[38] S. Aman, Y. Simmhan, and V. K. Prasanna, "Energy management systems: state of the art and emerging trends," *IEEE Communications Magazine*, vol. 51, no. 1, pp. 114–119, 2013.

[39] M. Amini, J. Frye, M. D. Ilić, and O. Karabasoglu, "Smart residential energy scheduling utilizing two stage mixed integer linear programming," in *North American Power Symposium (NAPS), 2015.* IEEE, 2015, pp. 1–6.

[40] Y.-M. Wi, J.-U. Lee, and S.-K. Joo, "Electric vehicle charging method for smart homes/buildings with a photovoltaic system," *IEEE Transactions on Consumer Electronics*, vol. 59, no. 2, pp. 323–328, 2013.

[41] K. Shimomachi, R. Hara, H. Kita, M. Noritake, H. Hoshi, and K. Hirose, "Development of energy management system for dc microgrid for office building:-day ahead operation scheduling considering weather scenarios," in *IEEE Power Systems Computation Conference*, 2014, pp. 1–6.

[42] L. Kelly, "Probabilistic modelling of plug-in hybrid electric vehicle impacts on distribution networks in british columbia," Ph.D. dissertation, University of Victoria, 2009.

[43] M. H. Amini, K. G. Boroojeni, C. J. Wang, A. Nejadpak, S. Iyengar, and O. Karabasoglu, "Effect of electric vehicle parking lots' charging demand as dispatchable loads on power systems loss," in *Electro Information Technology (EIT), 2016 IEEE International Conference on.* IEEE, 2016, pp. 0499–0503.

[44] M. Amini and A. Islam, "Allocation of electric vehicles' parking lots in distribution network," in *Innovative Smart Grid Technologies Conference (ISGT), 2014 IEEE PES*. IEEE, 2014, pp. 1–5.

[45] M. H. Amini, A. Kargarian, and O. Karabasoglu, "Arima-based decoupled time series forecasting of electric vehicle charging demand for stochastic power system operation," *Electric Power Systems Research*, vol. 140, pp. 378–390, 2016.

[46] A. M. Farid, "Symmetrica: test case for transportation electrification research," *Infrastructure Complexity*, vol. 2, no. 1, p. 1, 2015.

[47] Q. Wang, X. Liu, J. Du, and F. Kong, "Smart charging for electric vehicles: A survey from the algorithmic perspective," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1500–1517, 2016.

[48] R. Abousleiman and R. Scholer, "Smart charging: System design and implementation for interaction between plug-in electric vehicles and the power grid," *IEEE Transactions on Transportation Electrification*, vol. 1, no. 1, pp. 18–25, 2015.

[49] M. M. A. Abdelaziz, M. F. Shaaban, H. E. Farag, and E. F. El-Saadany, "A multistage centralized control scheme for islanded microgrids with pevs," *IEEE Transactions on Sustainable Energy*, vol. 5, no. 3, pp. 927–937, 2014.

[50] G. Carpinelli, F. Mottola, and D. Proto, "Optimal scheduling of a microgrid with demand response resources," *IET Generation, Transmission & Distribution*, vol. 8, no. 12, pp. 1891–1899, 2014.

[51] V. del Razo, C. Goebel, and H.-A. Jacobsen, "Vehicle-originating-signals for real-time charging control of electric vehicle fleets," *IEEE Transactions on Transportation Electrification*, vol. 1, no. 2, pp. 150–167, 2015.

[52] F. Kennel, D. Görges, and S. Liu, "Energy management for smart grids with electric vehicles based on hierarchical mpc," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 3, pp. 1528–1537, 2013.

[53] Y.-T. Liao and C.-N. Lu, "Dispatch of ev charging station energy resources for sustainable mobility," *IEEE Transactions on Transportation Electrification*, vol. 1, no. 1, pp. 86–93, 2015.

[54] K. N. Kumar, B. Sivaneasan, and P. L. So, “Impact of priority criteria on electric vehicle charge scheduling,” *IEEE Transactions on Transportation Electrification*, vol. 1, no. 3, pp. 200–210, 2015.

[55] W. Kempton and J. Tomić, “Vehicle-to-grid power implementation: From stabilizing the grid to supporting large-scale renewable energy,” *Journal of power sources*, vol. 144, no. 1, pp. 280–294, 2005.

[56] M. H. Amini, M. P. Moghaddam, and O. Karabasoglu, “Simultaneous allocation of electric vehiclesˇŻ parking lots and distributed renewable resources in smart power distribution networks,” *Sustainable Cities and Society*, vol. 28, pp. 332–342, 2017.

[57] H. N. Nguyen, C. Zhang, and J. Zhang, “Dynamic demand control of electric vehicles to support power grid with high penetration level of renewable energy,” *IEEE Transactions on Transportation Electrification*, vol. 2, no. 1, pp. 66–75, 2016.

[58] H. N. Nguyen, C. Zhang, and M. A. Mahmud, “Optimal coordination of g2v and v2g to support power grids with high penetration of renewable energy,” *IEEE Transactions on Transportation Electrification*, vol. 1, no. 2, pp. 188–195, 2015.

[59] W. Su, J. Wang, and J. Roh, “Stochastic energy scheduling in microgrids with intermittent renewable energy resources,” *IEEE Transactions on Smart Grid*, vol. 5, no. 4, pp. 1876–1883, July 2014.

[60] N. Xu and C. Chung, “Uncertainties of ev charging and effects on well-being analysis of generating systems,” *IEEE Transactions on Power Systems*, to appear.

[61] H. Zhao and A. Burke, “An intelligent solar powered battery buffered ev charging station with solar electricity forecasting and ev charging load projection functions,” in *Electric Vehicle Conference (IEVC), 2014 IEEE International*. IEEE, 2014, pp. 1–7.

[62] Q. Jiang, M. Xue, and G. Geng, “Energy management of microgrid in grid-connected and stand-alone modes,” *IEEE transactions on power systems*, vol. 28, no. 3, pp. 3380–3389, 2013.

[63] L. Yang, J. Zhang, and D. Qian, "Risk-aware day-ahead scheduling and real-time dispatch for plug-in electric vehicles," in *IEEE Global Communications Conference*, 2012, pp. 3026–3031.

[64] C. Marmaras, M. Corsaro, E. Xydas, L. Cipcigan, and M. Pastorelli, "Vehicle-to-building control approach for ev charging," in *IEEE International Universities Power Engineering Conference*, 2014, pp. 1–6.

[65] G. B. Dantzig, "Linear programming under uncertainty," in *Stochastic programming.* Springer, 2010, pp. 1–11.

[66] J. Xiong, S. Liu, X. Wang, D. Wu, and H. Zeng, "Impact assessment of electric vehicle charging on distribution systems at neighborhood levels," in *28th IEEE Canadian Conference on Electrical and Computer Engineering*, 2015.

[67] Tesla website, Accessed February 3, 2018, "Tesla model s sepcification," Available: http://www.teslamotors.com/en_CA/charging#/basics.

[68] Honda website, Accessed February 3, 2018, "Honda fit sepcification," http://automobiles.honda.com/alternative-fuel-vehicles/.

[69] S. Beer, T. Gomez, D. Dallinger, I. Momber, C. Marnay, M. Stadler, and J. Lai, "An economic analysis of used electric vehicle batteries integrated into commercial building microgrids," *IEEE Transactions on Smart Grid*, vol. 3, no. 1, pp. 517–525, 2012.

[70] Open EI, Accessed February 3, 2018, "Commercial building load consumption," https://ec.europa.eu/clima/policies/strategies/2030_en,Accessed:January30,2018.

[71] C. Zhou, K. Qian, M. Allan, and W. Zhou, "Modeling of the cost of ev battery wear due to v2g application in power systems," *IEEE Transactions on Energy Conversion*, vol. 26, no. 4, pp. 1041–1050, 2011.

[72] University of Queensland, Accessed February 3, 2018, "Weather and local environment," http://www.uq.edu.au/solarenergy/pv-array/weather.

[73] A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on stochastic programming: modeling and theory.* SIAM, 2014, vol. 16.

[74] M. L. Minsky and S. A. Papert, *Perceptrons - Expanded Edition: An Introduction to Computational Geometry.* MIT press, 1987.

[75] H. S. Hippert, C. E. Pedreira, and R. C. Souza, "Neural networks for short-term load forecasting: A review and evaluation," *IEEE Transactions on Power Systems*, vol. 16, no. 1, pp. 44–55, 2001.

[76] W. Mai, C. Chung, T. Wu, and H. Huang, "Electric load forecasting for large office building based on radial basis function neural network," in *2014 IEEE PES General Meeting— Conference & Exposition.* IEEE, 2014, pp. 1–5.

[77] A. J. Kleywegt, A. Shapiro, and T. Homem-de Mello, "The sample average approximation method for stochastic discrete optimization," *SIAM Journal on Optimization*, vol. 12, no. 2, pp. 479–502, 2002.

[78] ISO New England, Accessed February 3, 2018, "Day-ahead hourly locational marginal price[online]," http://www.iso-ne.com/isoexpress/web/reports/pricing/-/tree/lmps-da-hourly.

[79] Hydro Ottawa, Accessed February 3, 2018, "Time-of-use price structure," https://hydroottawa.com/accounts-and-billing/residential/rates-and-conditions.

[80] IEM, Accessed February 3, 2018, "National weather information," http://mesonet.agron.iastate.edu.

[81] D. Bunn and E. D. Farmer, *Comparative models for electrical load forecasting*, 1985.

[82] R. Zhang, Z. Y. Dong, Y. Xu, K. Meng, and K. P. Wong, "Short-term load forecasting of australian national electricity market by an ensemble model of extreme learning machine," *IET Generation, Transmission & Distribution*, vol. 7, no. 4, pp. 391–397, 2013.

[83] S. A.-h. Soliman and A. M. Al-Kandari, *Electrical Load Forecasting: Modeling and Model Construction.* Elsevier, 2010.

[84] S. Fan and R. J. Hyndman, "Short-term load forecasting based on a semi-parametric additive model," *IEEE Trans. Power Systems*, vol. 27, no. 1, pp. 134–141, 2012.

[85] A. Harvey and V. Oryshchenko, “Kernel density estimation for time series data,” *International Journal of Forecasting*, vol. 28, no. 1, pp. 3–14, 2012.

[86] P. Atsawathawichok, P. Teekaput, and T. Ploysuwan, “Long term peak load forecasting in Thailand using multiple kernel Gaussian Process,” in *ECTI-CON*, 2014, pp. 1–4.

[87] J.-B. Fiot and F. Dinuzzo, “Electricity demand forecasting by multi-task learning,” *IEEE Trans. Smart Grid*, 2016.

[88] F. R. Bach, G. R. Lanckriet, and M. I. Jordan, “Multiple kernel learning, conic duality, and the SMO algorithm,” in *ICML*, 2004, pp. 6–13.

[89] J. Zhuang, I. W. Tsang, and S. C. Hoi, “Two-layer multiple kernel learning.” in *AISTATS*, 2011, pp. 909–917.

[90] H. Xia and S. C. Hoi, “MKBoost: A framework of multiple kernel boosting,” *IEEE Trans. on Knowledge and Data Engineering*, vol. 25, no. 7, pp. 1574–1586, 2013.

[91] S. J. Pan and Q. Yang, “A survey on transfer learning,” *IEEE Trans. Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[92] M. Gönen and E. Alpaydın, “Multiple kernel learning algorithms,” *Journal of Machine Learning Research*, vol. 12, pp. 2211–2268, 2011.

[93] Y. Freund and R. E. Schapire, “Experiments with a new boosting algorithm,” in *ICML*, 1996, pp. 148–156.

[94] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, “Boosting for transfer learning,” in *ICML*, 2007, pp. 193–200.

[95] D. Pardoe and P. Stone, “Boosting for regression transfer,” in *ICML*, 2010, pp. 863–870.

[96] Y. Yao and G. Doretto, “Boosting for transfer learning with multiple sources,” in *CVPR*, 2010, pp. 1855–1862.

[97] O. Chapelle, P. Shivaswamy, S. Vadrevu, K. Weinberger, Y. Zhang, and B. Tseng, “Boosted multi-task learning,” *Machine Learning*, vol. 85, no. 1-2, pp. 149–173, 2011.

[98] B. Wang and J. Pineau, "Online boosting algorithms for anytime transfer and multitask learning," in *AAAI*, 2015, pp. 3038–3044.

[99] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Annals of Statistics*, pp. 1189–1232, 2001.

[100] L. Mason, J. Baxter, P. Bartlett, and M. Frean, "Boosting algorithms as gradient descent in function space," in *NIPS*, 2000, pp. 512–518.

[101] S. Rosset, J. Zhu, and T. Hastie, "Boosting as a regularized path to a maximum margin classifier," *Journal of Machine Learning Research*, vol. 5, pp. 941–973, 2004.

[102] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition.* Springer New York, 2009.

[103] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big Data*, vol. 3, no. 1, pp. 1–40, 2016.

[104] M. T. Rosenstein, Z. Marx, L. P. Kaelbling, and T. G. Dietterich, "To transfer or not to transfer," in *NIPS 2005 workshop on transfer learning*, vol. 898, 2005, pp. 1–4.

[105] P. Bühlmann and T. Hothorn, "Boosting algorithms: Regularization, prediction and model fitting," *Statistical Science*, pp. 477–505, 2007.

[106] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," *ACM Trans. Intelligent Systems and Technology*, vol. 2, no. 3, p. 27, 2011.

[107] Y. Chen, P. B. Luh, C. Guan, Y. Zhao, L. D. Michel, M. A. Coolbeth, P. B. Friedland, and S. J. Rourke, "Short-term load forecasting: similar day-based wavelet neural networks," *IEEE Transactions on Power Systems*, vol. 25, no. 1, pp. 322–330, 2010.

[108] . OPENEI, Accessed February 3, "[online]http://en.openei.org/doe-opendata/dataset," 2017.

[109] V. François-Lavet, D. Ernst, and R. Fonteneau, "On overfitting and asymptotic bias in batch reinforcement learning with partial observability," *arXiv preprint arXiv:1709.07796*, 2017.

[110] D. Wu, H. Zeng, C. Lu, and B. Boulet, “Two-stage energy management for office buildings with workplace ev charging and renewable energy,” *IEEE Transactions on Transportation Electrification*, vol. 3, no. 1, pp. 225–237, 2017.

[111] Y. Ma, F. Borrelli, B. Hencey, B. Coffey, S. Bengea, and P. Haves, “Model predictive control for the operation of building cooling systems,” *IEEE Transactions on control systems technology*, vol. 20, no. 3, pp. 796–803, 2012.

[112] V. François-Lavet, D. Taralla, D. Ernst, and R. Fonteneau, “Deep reinforcement learning solutions for energy microgrids management,” in *European Workshop on Reinforcement Learning (EWRL 2016)*, 2016.

[113] S. Vandael, B. Claessens, D. Ernst, T. Holvoet, and G. Deconinck, “Reinforcement learning of heuristic ev fleet charging in a day-ahead electricity market,” *IEEE Transactions on Smart Grid*, vol. 6, no. 4, pp. 1795–1805, 2015.

[114] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, “Playing atari with deep reinforcement learning,” *arXiv preprint arXiv:1312.5602*, 2013.

[115] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, p. 529, 2015.

[116] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, “Continuous control with deep reinforcement learning,” *arXiv preprint arXiv:1509.02971*, 2015.

[117] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming.* John Wiley & Sons, 1994.

[118] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction.* MIT press Cambridge, 1998, vol. 1, no. 1.

[119] D. Ernst, P. Geurts, and L. Wehenkel, “Tree-based batch mode reinforcement learning,” *Journal of Machine Learning Research*, vol. 6, no. Apr, pp. 503–556, 2005.

[120] M. Riedmiller, "Neural fitted q iteration–first experiences with a data efficient neural reinforcement learning method," in *European Conference on Machine Learning*. Springer, 2005, pp. 317–328.

[121] L.-J. Lin, "Reinforcement learning for robots using neural networks," Ph.D. dissertation, Fujitsu Laboratories Ltd, 1993.

[122] J. Xiong, D. Wu, H. Zeng, S. Liu, and X. Wang, "Impact assessment of electric vehicle charging on hydro ottawa distribution networks at neighborhood levels," in *Electrical and Computer Engineering (CCECE), 2015 IEEE 28th Canadian Conference on*. IEEE, 2015, pp. 1072–1077.

[123] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[124] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International Conference on Machine Learning*, 2016, pp. 1928–1937.

[125] G. Dalal, E. Gilboa, and S. Mannor, "Hierarchical decision making in electricity grid management," in *International Conference on Machine Learning*, 2016, pp. 2197–2206.

[126] P. Jain, V. Batra, and S. Darak, "Improved hierarchical decision making policy for reliable and green electricity grid," in *Communication Systems & Networks (COMSNETS), 2018 10th International Conference on*. IEEE, 2018, pp. 450–453.