

**LOAD BALANCING FOR SMART GRID: CENTRALIZED AND
DISTRIBUTED APPROACHES**

by

Mushu Li

Bachelor of Engineering, UOIT, 2015

A thesis

presented to Ryerson University

in partial fulfillment of the

requirements for the degree of

Master of Applied Science

in the Program of

Electrical and Computer Engineering

Toronto, Ontario, Canada, 2017

©Mushu Li, 2017

AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A THESIS

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my dissertation may be made electronically available to the public.

Load Balancing for Smart Grid: Centralized and Distributed Approaches

Master of Applied Science 2017

Mushu Li

Electrical and Computer Engineering

Ryerson University

Abstract

As one of the greatest concerns in the context of smart grid, the load balancing problem is addressed by improving the electrical power efficiency and stability via scheduling power loads, thereby shaping the power demand into the desired pattern. The research explores the load balancing strategies to reduce the demand fluctuations in the smart grid systems. Centralized and decentralized load balancing methodologies are discussed. For centralized approaches, offline and online exact power allocation methods are investigated by utilizing the geometric water-filling (GWF) approach. Furthermore, decentralized load balancing problem is discussed at power distribution sub-network level. Electrical vehicle (EV) fleet among the neighbouring charging stations is considered. Load balancing for the whole grid is achieved by local optimization processes via Proximal Jacobian Alternating Direction Method of Multipliers (ADMM) technique. Overall, facilitated by our proposed strategies, the reliability of the electric grid can be enhanced.

Acknowledgements

Firstly, I would like to express my sincere gratitude to my supervisor, Dr. Lian Zhao, for providing the invaluable assistance, guidance, and encouragement in my study and research. I will never forget her kind support and worthy advice over the past two years.

I would also like to thank my defense committee members Dr. Sri Krishnan, Dr. Amir-naser Yazdani, Dr. Soosan Beheshti and Dr. Lian Zhao for taking the time and effort to review my work and provide me their insightful comments.

I extend my thanks to Dr. Peter He and Dr. Jie Gao for their contributions to my research. The thesis can not be made without their helpful academic support.

Contents

<i>Declaration</i>	ii
<i>Abstract</i>	iii
<i>Acknowledgements</i>	iv
<i>List of Tables</i>	viii
<i>List of Figures</i>	ix
<i>List of Abbreviations</i>	xii
1 Introduction	1
1.1 Background and Motivation	1
1.2 Literature Review	4
1.3 Objectives	5
1.4 Contributions	6
1.5 Thesis Outline	8
2 Centralized Load Balancing Applying Water-filling Approach	10
2.1 Water-filling Approach	11
2.1.1 Conventional Water-filling Approach	11
2.1.2 Geometric Water-filling (GWF) Approach	13
2.2 Problem Statement	15
2.3 Basic Load Allocation Scheme	18
2.4 Offline Elastic Load Power Allocation	22

2.5	Performance Evaluation	23
2.6	Chapter Summary	27
3	Dynamic Centralized Load Balancing Implementation	29
3.1	System Model for Online Approach	30
3.1.1	Network Structure	30
3.1.2	Problem Formulation	33
3.2	Online Elastic Load Power Allocation	33
3.3	Computation Efficient Online Approach	35
3.4	Performance Evaluation	37
3.5	Chapter Summary	42
4	Decentralized Load Balancing by EV Fleets	44
4.1	Overview of ADMM	47
4.2	System Model	49
4.3	Decentralized EV Power Allocation Optimization Schemes	52
4.3.1	Basic Decentralized EV Power Allocation	53
4.3.2	Dynamic Decentralized EV Power Allocation	60
4.4	EV Random Charging Scheme: Implementation of Optimal Power Allocation	65
4.5	Performance Evaluation	69
4.5.1	Simulation Results of Basic Decentralized EV Power Allocation Scheme	69
4.5.2	Simulation Results of Dynamic Decentralized EV Power Allocation Scheme	71
4.6	Chapter Summary	74
5	Conclusion and Future Works	76
5.1	Conclusion	76
5.2	Future Works	77

List of Tables

2.1	List of variables for centralized load balancing	11
2.2	Household Appliances Parameter Setting	24
2.3	Simulation Results for Offline ELPA, SDWF, and Unscheduled Case	26
3.1	Standard Deviation of Overall Power Consumption in 5 Consecutive Days (Fig. 3.3 and 3.4).	41
4.1	List of variables for centralized load balancing	46

List of Figures

1.1	The power demand profile during the day of Aug. 1, 2016 in Ontario [1]. . .	2
1.2	The relationship among Chapter 2 to 4 for solving the load balancing problem.	9
2.1	Visualized the conventional water-filling approach.	13
2.2	Illustration for the geometric water-filling approach.	13
2.3	System model for the centralized load balancing problem.	16
2.4	Illustration of the basic load allocation scheme.	20
2.5	(a) Elastic load allocation among time slots. (b) Elastic load allocation among groups. (c) Elastic load allocation among users in a group.	24
2.6	Power consumption of inelastic, elastic, and total loads scheduled by offline ELPA for three days.	25
2.7	Free-run (Unscheduled) total power consumption and the power consumption achieved by Offline ELPA and SDWF for three consecutive days.	26
2.8	Power consumption of inelastic, elastic, and total loads, where PU is 2000 kWh.	27

2.9	(a) Elastic load power consumption in group view, where PG is infinity in five hours, 30 minutes in a time slot. (b) The power consumption of inelastic, elastic, and total loads, where PG is infinity in a day. Legend is the same as Fig. 2.6. (c) Elastic load power consumption in group view, where Group 4 PG is bounded in 600 kWh in five hours, 30 minutes in a time slot. (d) The power consumption of inelastic, elastic, and total loads, where Group 4 PG is bounded in 600 kWh in a day. Legend is the same as Fig. 2.6.	28
3.1	Network model for the centralized load balancing problem.	31
3.2	Moving Window Model	32
3.3	Overall power consumption scheduled by OELPA and unscheduled power consumption for 5 days. (a) With same load budget. (b) Load budget increasing. (c) Load budget descending.	39
3.4	Power consumption of overall loads scheduled by EOELPA and OELPA for 5 days. (a) With same load budget. (b) Load budget increasing. (c) Load budget descending.	40
3.5	Standard deviation vs. parameter m for unscheduled, OELPA, and EOELPA schemes.	42
4.1	Power profile for (a) a residential unit, (b) a retail unit, and (c) an industrial unit in a month with power base [kWh/(24*30)] [2].	45
4.2	System model for EV charging in neighbouring charging stations.	50
4.3	Illustration for Algorithm 7, where $E_2 = \sum_{e \in \chi_2} P_c T_{min_e}$	59
4.4	Illustration for Algorithm 8, where $P_{2,k} = \sum_{e \in \chi_{2,k}} P_c$	66
4.5	EV charging scheme flow chart.	67
4.6	The blocking probability $pb(\varphi_{g,k})$ vs. $\varphi_{g,k}$	69

4.7	The power demand profile for EV loads and base loads among 5 buses and the whole system by the basic decentralized EV power allocation scheme in one day. <i>PG</i> is infinity.	71
4.8	The power profile for EV loads and base loads among 5 buses and the whole system by Proximal Jacobian ADMM and random charging scheme in one day with finite <i>PG</i>	72
4.9	The comparison of overall power demand scheduled without EV fleeting, with EV fleeting and <i>PG</i> constraint, and without <i>PG</i> constraint.	73
4.10	The overall power profile for 5 buses and the whole system for unscheduled and scheduled cases in 6 days.	74
4.11	The power profile for EV loads and base loads among 5 buses and the whole system by the dynamic EV load allocation scheme in one day.	75

List of Abbreviations

ADMM Alternating Direction Method of Multipliers

BLA Basic Load Allocation

DSM Demand Side Management

ELPA Elastic Load Power Allocation

EOELPA Computation Efficient Online Elastic Load Power Allocation

ESMAP Energy Sector Management Assistance Program

EV Electrical Vehicle

GWF Geometric Water-filling

HVAC Heating, Ventilation, and Air Conditioning

IoT Internet of Things

OELPA Online Elastic Load Power Allocation

PHEV Plug-in Hybrid Electrical Vehicle

RRM Radio Resource Management

SOC State of Charge

Chapter 1

Introduction

1.1 Background and Motivation

As the conventional electricity grid shows the difficulty in supplying the increasing power demand nowadays, the concept of smart grid has been evolved to improve the performance of power grid. In the definition of smart grid, the enhanced electricity delivery network is established with integrating the advanced technologies, such as intelligent and autonomous controllers, data management, and communication between utilities and consumers [3]. With smart grid, an efficient and reliable delivery system can be accomplished by energy monitoring, energy modeling, and strategy adjustments [4].

It is well known that the power demand fluctuates in a day, and it could swing significantly by the time [5]. Fig. 1.1 shows the demand profile during a typical summer weekday in Ontario. The power demand during peak demand hours could be as high as 148% of the demand in non-peak hours. The rise of energy demand brings inefficiency and instability for the power generation and delivery: firstly, to match the high demand, the utilities have to invest more standby generators which will not operate during non-peak hours. Both utilities and consumers share the expensive cost brought from extra generators. In power grids allocated in USA Eastern Interconnection, \$28 billion could be saved with 10% demand cut

out [6]. On the other hand, the power shortage could happen if the high electric demand can not be met in peak demand hours. To solve the above problems, Demand Side Management (DSM) is proposed to manage the power load with appropriate strategies. As an analytical technique in smart grid, DSM provides a mechanism to shape the power demand in the grid by scheduling power loads to the desired demand pattern [7]. With DSM, elastic loads, which have flexibility on the time and power, could be shifted to fill the demand gap in non-peak hours for the purpose of reducing power fluctuation, such as PHEVs (Plug-in Hybrid Electrical Vehicles) load and HVAC (Heating, Ventilation, and Air Conditioning) load. Whereas, the loads, which are strictly on the time and power, are defined as inelastic loads [5,8]. By strategies of load scheduling by DSM, the elastic load power can be managed to achieve load balancing in the smart grid.

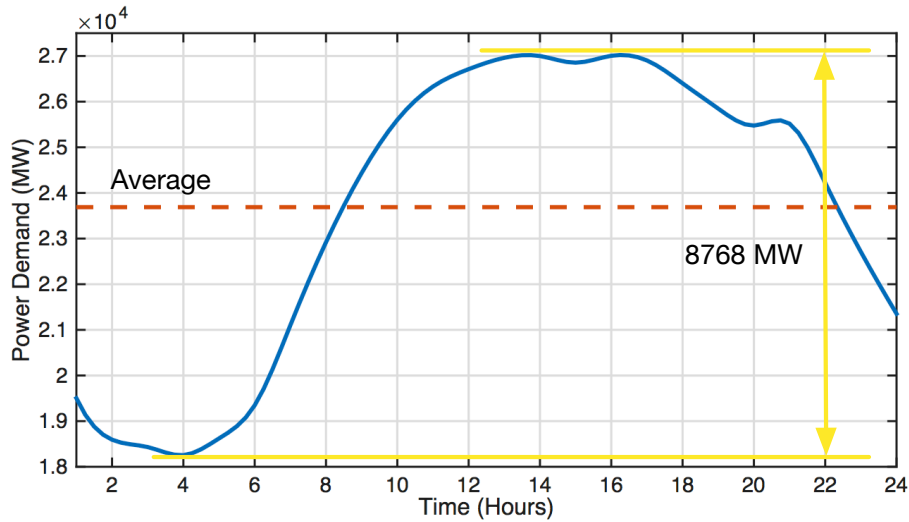


Figure 1.1: The power demand profile during the day of Aug. 1, 2016 in Ontario [1].

The mechanism of DSM is normally classified by several categories. For the network structure, the centralized and decentralized load management structures are utilized for the control mechanism. By the centralized manner, users communicate with controller directly without sharing the information with other users. By this structure, the optimal strategy for the global system is computed by the central controller, and users operate the system with the

strategy given by the controller. The drawback of the centralized structure is the mass data exchanging and higher computation in the controller. In the decentralized structure, users share the computation burden locally and interact the information with other users to achieve the optimal global solution. In the aspect of load scheduling objectives, the price-based method is aiming for minimizing the electrical cost via managing power demand regarding the time-varying electricity price. While, in the incentive-based method, the demand is scheduled to reduce the electric stress in the fixed power payment with the corresponding reward and penalty [9]. The method proposed in this work is designed to flatten overall power demand, which could improve the grid efficiency and seemingly uniform the price scheme [5].

The demand side management has been practiced in real power grids in recent years. The government of Queensland, Australia, has established Energy Conservation and Demand Management Program to regulate the residential and commercial loads since 2011 [10]. The energy management center centralizes the load management to schedule the load usage at peak times. Furthermore, the demand side management provides a solution for the energy limitation problem. Energy Sector Management Assistance Program (ESMAP) is the program to manage the power loads in a centralized manner by installing intelligent load controllers for some villages allocated in the developing countries, where the electric power is limited [11]. For the power grid in Ontario, the power demand management is achieved by different time-of-use rate, which is depending on when customers use electricity [12]. Customers will be charged with higher electric price in the peak hours, and enjoy the less electric price in non-peak hours. Also, the increasing popularity of PHEV loads provides plenty of elastic power to be scheduled in the recent years. Studies in [13, 14] remind that if 10% of vehicles on the road are electric, the power grid will show difficulty to ensure the power quality at the low-voltage (LV) level in peak demand hours. The effective PHEV charging scheme is required in urgent not only for the purpose to flatten the demand but also for stabilizing the power distribution systems.

1.2 Literature Review

Load balancing has attracted lots of researchers' interests in the smart grid field. The elastic loads had been taken advantage to response the demand management in [15–19] based on the real-time electric price. Via controlling the elastic loads, the utility of users related to the current price is maximized for the global network in these works. The works in [20, 21] implemented the real-time price control to minimize the power fluctuation by concerning the users' payoff and the utilities' profits maximization. The optimization tools are also implemented to solve load balancing problem [5, 22–26].

PHEV and Electric Vehicles (EV) are typical elastic loads which have high flexibility on the charging time and power. As the environmental concern and high fuel rate, PHEVs are expected to be widely used in the future. The works [25, 27–32] showed that PHEV load allocation is an efficient methodology to smooth the overall power demand. In [31, 33], charging strategies for PHEVs in charging stations were proposed for a large number of PHEVs plugged in the system. Furthermore, since the mobility of PHEV loads, it is also considered as a mechanism to balance the power demand across multiple distribution sub-networks. The EV fleets and aggregators are considered to balance the loads in [34–39] due to the mobility of the EVs. An Alternating Direction Method of Multipliers (ADMM) method is applied for the distributed EV charging control in [34, 35, 40]. The EV aggregator method is proposed and fits into the exchange optimization problem in [34]. EVs are controlled individually to achieve the corresponding global objective, such as valley-filling problem and price-based problem. The performance of load balancing by EV fleets was evaluated in [36]. The EV fleets were paired to fit the symmetrical model and balanced the demand in multiple districts by achieving synchronous stability in the system. Moreover, [37] aimed to reduce the overall power cost through PHEV fleets across distribution sub-networks by optimizing an NP-hard problem. The work [38] also proposed a PHEV charging strategy among neighbouring charging stations by a consensus model in the decentralized manner to

manage power demand.

Water-filling approach is a significant optimization tool for Radio Resource Management (RRM) to maximize channel capacity under the power constraint in communication systems [41, 42]. The water-filling approach is also an optimization tool for power demand management. The early work [5] scheduled the elastic loads for household appliances in residential units, such as PHEV and cloth washer, with the assumption that all the load information is available. The proposed algorithm utilized the water-filling approach to flatten the overall power consumption. Furthermore, [25] developed several algorithms applying the water-filling concept into PHEVs charging problem. With those algorithms, the demand peak brought by PHEVs enabled to be shaved, and PHEV loads were scheduled in nonpeak hours to fill the demand gap. Water-filling approach is also applied in HVAC application related with the temperature adjustment in [24]. HVAC load operated with accompanying load management strategy, and the users' comfort level would not be influenced. The centralized controlling cooperating with the distributed controllers is modeled in the work. The works [23, 43] also analyzed the impact of the delay cost on the optimality of the water-filling approach.

1.3 Objectives

In this study, the effective load balancing strategy is discussed to reduce the power fluctuation for the entire electricity grid and distribution sub-networks by scheduling the elastic load demand. The elastic loads in peak hours are shifted to the nonpeak hours with the proposed strategies. Overall power fluctuation is significantly reduced subject to upper bound constraints. The centralized and decentralized load balancing approaches are presented in this study respectively. The load management in three levels, which include the whole grid, groups, and users, is considered in the centralized method. Meanwhile, load balancing among the distribution sub-networks is analyzed in the decentralized method, with which

the computation burden is transferred to local distribution sub-networks.

The centralized load balancing problem will be introduced in the offline and online scenarios. The geometric water-filling (GWF) approach, which proposed in [41], is applied as an optimization tool to shape the power demand with considering sum power constraints. In the offline environment, all the load information is predicted by the central controller in a day. The central controller will gather the load information, and manage the elastic load power referring to the peak power constraints for the whole grid. On the other hand, the online solution is developed from the previous approach to schedule the elastic loads without the knowledge of the future load demand. The dynamic controlling is applied to reduce the influence of the failed prediction. The computation efficient online algorithm is proposed to reduce the computation and communication load for the real-time management in the online solution.

Besides the centralized solution, the decentralized elastic load management among distribution sub-networks is presented in the research. EV load is utilized as a significant role for load balancing among sub-networks because of the mobility it has. The solution of EV load management in neighbouring charging stations is proposed. EVs will be charged in a specific charging station to fill the demand gap in the sub-network to minimize the power fluctuation for the whole system. Proximal Jacobian Alternating Direction Method of Multipliers (ADMM) technique is used to allocate the EV power in a decentralized manner, and EV loads are scheduled by a random access method scheme to implement the given power allocation strategy.

1.4 Contributions

The main contribution of this research is summarized as follows:

1. The water-filling approach is applied to balance the power demand in offline and online scenarios for centralized demand side management. The upper bounds of the power

demand are considered to avoid the overload on the transformers in groups or user nodes. By the water-filling approach, the exact solution can be solved with low degree computational complexity. The performance improvement is demonstrated by comparing the benchmark work presented in [5]. The exact load balancing solution by the water-filling approach is published in [44].

2. The dynamic load balancing approach provides a robust algorithm independent with the predicted power demand. It reduces the influence from the failure prediction and balances the power demand with adopting varied demand budget in real-time. A two-way communication network among the smart grid operator, groups, and users is established for operating the proposed approach. The smart grid operator will adjust the reference level with the feedback from user-ends, *i.e.*, smart meters, in every communication cycle, such that the power allocation can be more stable compared with the conventional approach with one direction communication from the smart grid operator to end users. Furthermore, the burden of communication and computation brought from the dynamic approach is reduced by the proposed computation efficient algorithm. The research work published in this aspect includes [45, 46].
3. The optimization tool, water-filling approach, is developed to solve the problem from the Radio Resource Management (RRM) to load balancing for smart grid in this work. The applications of the water-filling approach are expanded by exploring the common points between two problems: the parameterized objective power demand and the constrained power to be allocated. Various optimization tools could be employed in load balancing problem for possible solutions, such as Semi-Markov Decision Process (SMDP). The flexible applications of SMDP are explored in the published works [47, 48].
4. The decentralized algorithm for load balancing among distribution sub-networks is introduced in the study. By Proximal Jacobian Alternating Direction Method of Multipliers (ADMM), the whole system optimization can be achieved by local computation

on EV charging allocation. The information sharing among the sub-networks is reduced significantly. The proposed method reduces the computation and communication load for the sub-networks and enables the processing of the demand management in the large scale systems. The peak shaving in the sub-networks is achieved by EV fleets while satisfying the EV fleets' charging requirements. The accepted paper [49] reported the proposed decentralized load balancing scheme.

1.5 Thesis Outline

The remainder of the thesis is organized as follows. In Chapter 2, the centralized load balancing by the water-filling approach is introduced. The system model of the centralized problem is presented, and the offline elastic load allocation scheme is proposed. The scheme enables to solve the load balancing problem with full prediction of the power information. Simulation results for the offline scenario are provided by the comparison with the result of the benchmark work [5] in this chapter.

Chapter 3 investigates the dynamic power allocation developed from the previous chapter. The real-time solution for load balancing without the knowledge of the future information is proposed in this chapter. Furthermore, the computation efficient online approach is introduced to reduce the computation burden. Simulation results for the online approach are presented.

Chapter 4 provides the decentralized load balancing for sub-networks in details. This chapter elaborates the system model for EV charging scheduling among neighbouring charging stations for both offline and online environment. The decentralization solution and simulation results are presented in this chapter.

Finally, Chapter 5 concludes the thesis. The chapter also discusses the future works for the study.

The relationship among Chapters 2 to 4 is shown in Fig. 1.2. Firstly, the basic elastic

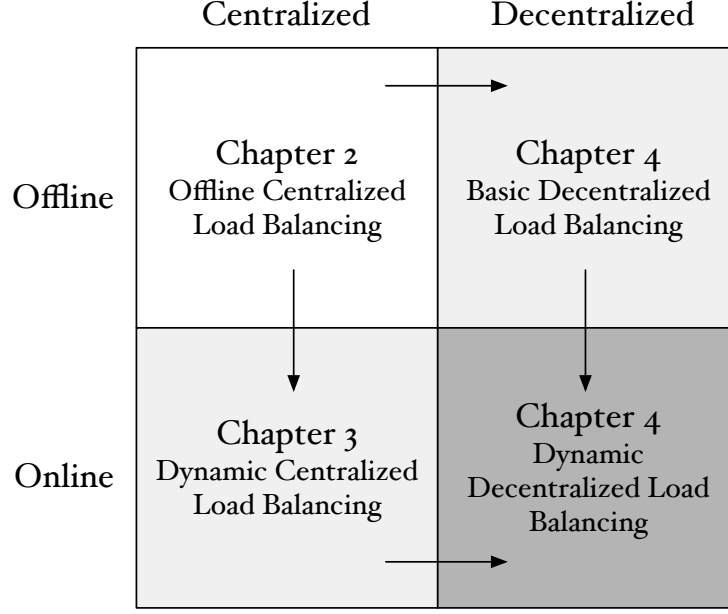


Figure 1.2: The relationship among Chapter 2 to 4 for solving the load balancing problem.

load allocation is proposed to solve the offline load allocation problem in Chapter 2. Then, with the same objective function but accommodating the dynamic loads, the online load allocation is developed to solve the load balancing problem without predicted future load information in Chapter 3. Furthermore, in Chapter 4, the basic decentralized load balancing is formulated by the similar objective of offline load allocation scheme, while it achieves distributed load balancing regarding distribution sub-networks. Finally, the dynamic decentralized load balancing strategy is evolved from the objectives from the basic decentralized scheme and dynamic centralized scheme, such that the new proposed scheme can balance the power load without future information in a decentralized manner. Generally, centralized load balancing schemes are introduced in Chapter 2 and 3 for offline and online scenarios respectively. Chapter 4 presents the offline and online schemes to balance the loads in a decentralized manner.

Chapter 2

Centralized Load Balancing Applying Water-filling Approach

In this chapter, the three-dimensional system structure is formed to achieve centralized load balancing in smart grid. All the load information is predicted, including the elastic and inelastic load information of all the user nodes. To simplify the problem, the two-dimensional solution is formulated by geometric water-filling (GWF) approach introduced in [41]. The offline elastic load power allocation scheme (ELPA) is developed by applying the 2-D solution to allocate elastic load power in the 3-D structure. Simulation results are presented to depict the achieved load balancing performance. In the centralized load balancing problem discussed in this work, we assume that elastic loads are sufficient to balance the loads, and elastic loads are available to be scheduled for the whole day, which is the same as the assumption made in [5]. Table 2.1 is a list of the variables used in analysis of the centralized load balancing problem.

Table 2.1: List of variables for centralized load balancing

Group	Variable	Meaning
Index Group	k	index of time slots, for $k = 1, \dots, K$.
	i	index of group, for $i = 1, \dots, I$.
	j	index of user (or node), for $j = 1, \dots, J$.
	χ_i	the i th group.
	n	index of time slots in the reference window, for $n \in [k - N + 1: k]$.
	n_1	index of time slots in past part of the reference window, for $n_1 \in N_1$, where $N_1 = [k + m - N + 1: k]$.
	n_2	index of time slots in future part of the reference window, for $n_2 \in N_2$, where $N_2 = [k + 1: k + m]$.
Load Group	\mathbf{A}	inelastic load matrix. The (k,j) th element, $A_{k,j}$, denotes inelastic load of user j at time k .
	\mathbf{B}	inelastic load vector, obtained by sums of \mathbf{A} over rows. The k th element, B_k , denotes the total inelastic load at time k .
	\mathbf{R}	elastic load matrix. The (k,j) th element, $R_{k,j}$, denotes inelastic load of user j at time k .
	\mathbf{S}	elastic load vector. The k th element, S_k , denotes the total elastic load at time k .
	$\hat{\mathbf{A}}$	predicted inelastic load matrix. The (k,j) th element, $\hat{A}_{k,j}$, denotes predicted inelastic load of user j at time k .
Constraint Group	\mathbf{P}	individual load upper bound (or peak) matrix. The (k,j) th element, $P_{k,j}$, denotes load upper bound for user j at time k .
	\mathbf{PG}	group load upper bound matrix. The (k,i) th element, $PG_{k,i}$, denotes load upper bound for the i th group at time k .
	\mathbf{PU}	load upper bound vector for time slots. The k th element, PU_k , denotes load upper bound at time k .
	P_T	total load budget or supply capability for entire elastic power loads over all the time slots, <i>i.e.</i> , a sum of loads.
	\mathbf{P}_S	elastic load demand vector. The k th element, P_{S_k} , denotes the elastic load demand at time k .
Reference Level Group	L	reference horizon, reference level for offline scheduling.
	\mathbf{L}	reference level vector for online scheduling. The k th element, L_k , denotes the reference level at time k .

2.1 Water-filling Approach

2.1.1 Conventional Water-filling Approach

Water-filling approach has been widely used for the power allocation of Radio Resource Management (RRM) problems in the communication systems. As a well-known optimization tool [41, 42, 50, 51], it is maximizing the mutual information between the input and output

among multiple subchannels with the total power constraint [42]. The classic water-filling result is the solution of the following optimization problem:

$$\begin{aligned} \max_{\{s_i\}_{i=1}^K} \quad & \sum_{i=1}^K \log(1 + a_i s_i) \\ \text{subject to} \quad & s_i \geq 0, \forall i; \\ & \sum_{i=1}^K s_i = P, \end{aligned} \tag{2.1}$$

where a_i and s_i are the channel gain and allocated power for the i th channel respectively. The power allocated in any subchannel is non-negative, and the overall power is equal to P . The parameter l is denoted as the water-level. Then, the solution is given by the following nonlinear equation set:

$$\begin{cases} s_i = (l - \frac{1}{a_i})^+, \quad \forall i \\ \sum_{i=1}^K s_i = P, \\ l \geq 0, \end{cases} \tag{2.2}$$

where $(l - \frac{1}{a_i})^+ = \max\{0, (l - \frac{1}{a_i})\}$, and the water level is a positive parameter to satisfy the sum power constraint. Equations in Eq. (2.2) are illustrated in Fig. 2.1. The water-filling problem can be visualized as a water-tank. The reciprocal of the channel gain for subchannels is visualized by the steps in the bottom of the water tank. The solution for the objective function can be interpreted by pouring certain amount of water into the tank. The shadowed area indicates the water volume after filling the water in the tank. The optimal allocated power for subchannels is denoted by the water volume on top of those steps.

The solution for the conventional water-filling problem is presented in [42]. The iterative search method is applied to obtain the water-level l , and the allocated power can be derived from Eq. (2.2). For the best sorting algorithm for the conventional water-filling algorithm, the complexity is $O(K \log_2 K)$ [42]. The worst-case complexity is $O(K^2)$ [41].

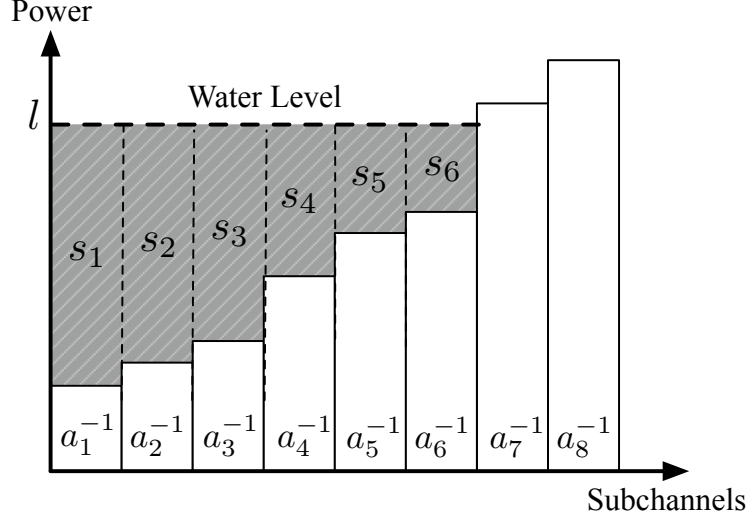


Figure 2.1: Visualized the conventional water-filling approach.

2.1.2 Geometric Water-filling (GWF) Approach

The geometric water-filling (GWF) approach proposed in [41] solves the water-filling problem in geometric view, such that it avoids the water-level searching by solving the nonlinear equation set. The GWF algorithm is illustrated in Fig. 2.2.

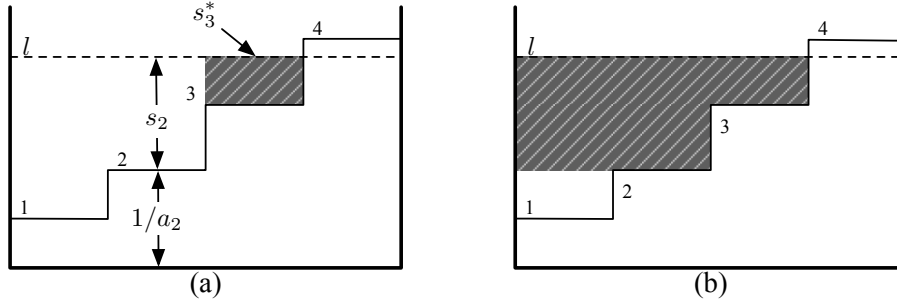


Figure 2.2: Illustration for the geometric water-filling approach.

Suppose there are 4 subchannels to allocate power, *i.e.*, $K = 4$, with unit step width in the water tank. Rather than searching the water level l as the first step in the conventional water-filling approach, GWF determines the water level by finding the highest step under

the water. The step depth represents the inverse of the channel gain, $1/a_i$, and the steps are sorted in monotonically increasing order. The subchannel with the highest step depth under the water is denoted as k^* . To define the mathematical expression, the water volume above the step k is defined as $P_2(k)$. For example, in Fig. 2.2(b), the shadow area denotes the $P_2(k = 2)$. The water volume $P_2(k)$ can be determined by the overall water volume constraint P and the water volume under the step as the following equation:

$$P_2(k) = \left\{ P - \left[\sum_{k'=1}^{k-1} \left(\frac{1}{a_k} - \frac{1}{a_{k'}} \right) \right] \right\}^+, k = 1, \dots, K. \quad (2.3)$$

Then the index k^* denotes the highest step under water. It can be expressed as the maximum index number with nonzero $P_2(k)$ value,

$$k^* = \max\{k | P_2(k) > 0, 1 \leq k \leq K\}. \quad (2.4)$$

And the power allocated for the step k^* is determined as following:

$$s_{k^*} = \frac{1}{k^*} P_2(k^*). \quad (2.5)$$

The power allocation for other steps can be determined from s_{k^*} and the step depth difference. The following equation shows the explicit solution to problem (2.1):

$$s_k = \begin{cases} s_{k^*} + \left(\frac{1}{a_{k^*}} - \frac{1}{a_k} \right), & 1 \leq k \leq k^* \\ 0, & k^* < k \leq K. \end{cases} \quad (2.6)$$

The solution for GWF is brought from Eq. (2.3) to (2.6). With GWF, the water-filling problem can be simplified without non-linear searching water level processes. GWF provides an exact optimal solution with low degree polynomial computational complexity [41], and the optimality is proven [41].

By the above solution, the water-filling algorithm is shown as a powerful tool to solve the problem of channel capacity maximization. It deals with power allocation process with the constraint on sum power, and the allocated power is non-negative. The channel quality is served as the parameter in the optimization problem, which is visualized as fixed steps inside the water tank. The water-level is introduced as the parametrized value to satisfy the sum power constraints. The common points exist between power allocation in the communication application and the power allocation in smart grids. For the load balancing problem discussed in this work, the objective is minimizing power fluctuation with the finite power budget among time slots. The overall elastic load power budget can be regarded as the total water volume pouring into the water tank. The inelastic load power, which can not be scheduled, can be regarded as the fixed steps in the water tank. The power allocation over time units could be achieved by calculating the water volume above those steps. The allocated power demand in time slots is aiming to achieve the defined water level, referred as “reference level” in the remaining of the thesis. The objective of load balancing problem can be interpreted as balancing the overall power demand for the grid around the reference level by elastic loads allocation. The research work focuses on a novel application of load balancing in smart grid with employing the communication optimization tool. This research presents the extended application of optimization tools from the wireless communication network, and it shows that the water-filling is a desirable tool for optimizing the load scheduling to achieve load balancing.

2.2 Problem Statement

There are two categories of power loads considered in this problem: elastic loads and inelastic loads. The objective of this chapter is to allocate the elastic load power to flatten the energy consumption in the offline scenario. The inelastic load powers serve as a sequence of constants since they have to be satisfied as requested. Fig. 2.3 shows a three-dimensional

model regarding the time slots, the load powers, and the users. Assuming K time slots are considered, and k is the index of the time slots. All the load information is predicted among those time slots. There are J users participating the power allocation in the model, which are grouped into I groups or distribution sub-networks. The indexes, j and i , are the indexes of users and groups respectively. The set, χ_i , is the set of the users in group i .

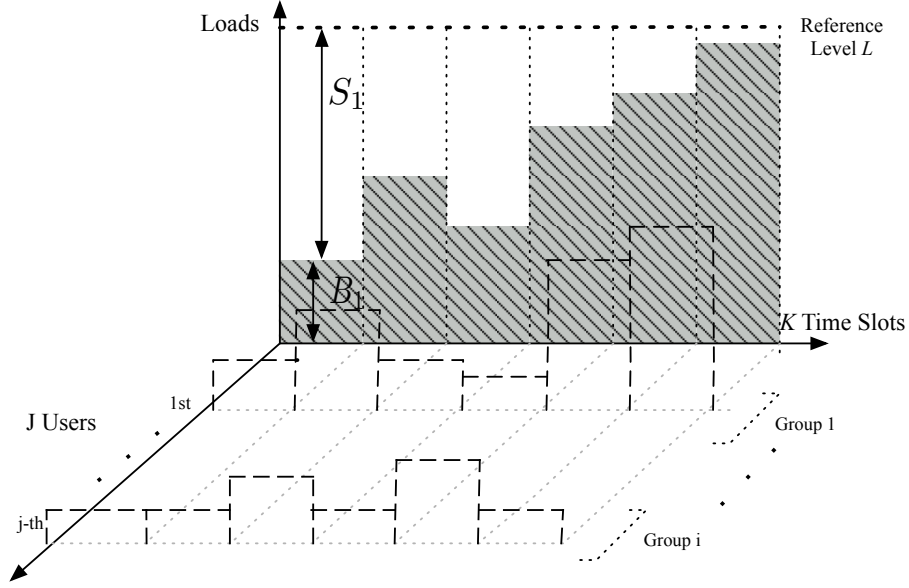


Figure 2.3: System model for the centralized load balancing problem.

The reference level L is an optimization variable from the water-filling concept, which indicates the objective of the overall power consumption over time slots. The users' inelastic load power and elastic load power are denoted as matrix \mathbf{A} and \mathbf{R} , where $A_{k,j}$ and $R_{k,j}$ denote the inelastic and elastic load power of user j in the time slot k , respectively. In Fig. 2.3, the projection of inelastic and elastic load power for the whole grid in the k th time slot is indicated as the shadowed area B_k and S_k respectively, where:

$$\sum_{i=1}^I \sum_{j \in \chi_i} R_{k,j} = S_k; \quad \sum_{i=1}^I \sum_{j \in \chi_i} A_{k,j} = B_k. \quad (2.7)$$

The objective of load balancing with predicted load information is minimizing the devi-

ation of the overall power consumption for each time slot from the reference level L across the observation interval. The objective function can be summarized as follows:

$$\begin{aligned}
& \min_{\{\mathbf{R}, L\}} \quad \sum_{k=1}^K \left[\sum_{i=1}^I \sum_{j \in \chi_i} (A_{k,j} + R_{k,j}) - L \right]^2 \\
& \text{subject to} \quad 0 \leq R_{k,j} \leq P_{k,j}, \forall j, k; \\
& \quad \sum_{j \in \chi_i} R_{k,j} \leq PG_{k,i}, \forall i, k; \\
& \quad \sum_{j=1}^J R_{k,j} \leq PU_k, \forall k; \\
& \quad \sum_{k=1}^K \sum_{j=1}^J R_{k,j} = P_T; \\
& \quad L \geq 0.
\end{aligned} \tag{2.8}$$

In the objective function (2.8), the mean square errors between the overall power demand and the reference level L is expected to be minimized among time slots to achieve load balancing for the whole power grid. The elastic load allocation and the reference level are determined in Eqn. (2.8), where $P_{k,j}$, $PG_{k,i}$, and PU_k are the elastic power upper bound for user node j , group i , and the whole grid in the k th time slot. The term, P_T , is the overall elastic load budget allocated in the system. All the upper bound constraints should be non-negative to ensure the feasibility for power allocation. To ensure obtaining a feasible solution, the overall power budget should be enabled to be allocated within the constrained power, no matter for the user node upper bound \mathbf{P} , the group upper bound \mathbf{PG} or the overall upper bound in time slot \mathbf{PU} . Therefore, the upper bounds, \mathbf{P} , \mathbf{PG} , and \mathbf{PU} , are assumed as follows:

$$\begin{cases} \sum_{k=1}^K \sum_{j=1}^J P_{k,j} \geq P_T, \\ \sum_{k=1}^K PU_k \geq P_T, \\ \sum_{k=1}^K \sum_{i=1}^I PG_{k,i} \geq P_T. \end{cases} \tag{2.9}$$

If above conditions can not be met, then

$$P_T = \min \left\{ \sum_{k=1}^K \sum_{j=1}^J P_{k,j}, \sum_{k=1}^K PU_k, \sum_{k=1}^K \sum_{i=1}^I PG_{k,i} \right\}. \quad (2.10)$$

Similarly, for constraints \mathbf{P} , \mathbf{PG} , and \mathbf{PU} , the assumptions are shown as follows:

$$\begin{cases} \sum_{j \in \chi_i} P_{k,j} \geq PG_{k,i}; \text{ otherwise, } PG_{k,i} = \sum_{j \in \chi_i} P_{k,j}, \\ \sum_{i=1}^I PG_{k,i} \geq PU_k; \text{ otherwise, } PU_k = \sum_{i=1}^I PG_{k,i}. \end{cases} \quad (2.11)$$

The offline elastic power allocation can be summarized in the following steps: Firstly, the power demand is forecasted by the smart grid operator based on previous demand behavior, including the inelastic load power \mathbf{A} for all the users and the overall elastic load power budgets P_T for all the time slots. Then, the reference level L is determined by the smart grid operator after gathering all the load information. The elastic load power \mathbf{S} can be allocated with the appropriate algorithm. In this step, the reference level could be adjusted depending on the constraints. Afterwards, the elastic loads are filled into the elastic power \mathbf{S} which were analyzed in previous steps.

2.3 Basic Load Allocation Scheme

To simplify Eqn. (2.8), a two-dimensional load balancing problem is introduced via analyzing the overall power demand for all the users among the time slots, where S_k and B_k represent the elastic load power and inelastic load power for the whole power grid in the k th time slot, respectively. Then the objective function (2.12) below expects to minimize the fluctuation between the overall demand and the reference level across studied time span, from the first

to the K th time slot, as:

$$\begin{aligned}
& \min_{\{S_k, L\}} \quad \sum_{k=1}^K (S_k + B_k - L)^2 \\
& \text{subject to} \quad 0 \leq S_k \leq PU_k, \forall k; \\
& \quad \sum_{k=1}^K S_k = P_T; \\
& \quad L \geq 0.
\end{aligned} \tag{2.12}$$

Since the objective function is similar to the sum rate maximizing problem in [41], the geometric water-filling approach can be applied to solve the problem (2.12). To ensure the optimality for GWF, the Lagrange function of (2.12) is:

$$\mathcal{L}(S_k, B_k, L, \gamma_k, \lambda, \phi) = \sum_{k=1}^K (S_k + B_k - L)^2 - \sum_{k=1}^K \gamma_k S_k - \lambda \left(\sum_{k=1}^K S_k - P_T \right) - \phi L, \tag{2.13}$$

where γ_k , λ , and ϕ are dual variables for constraints. To solve the powers and reference level L , the Karush-Kuhn-Tucker (KKT) conditions are:

$$\begin{cases} 2 \sum_{k=1}^K (S_k + B_k) - 2KL + \phi = 0; \\ L \geq 0, \quad \phi \geq 0; \\ L \cdot \phi = 0. \end{cases} \tag{2.14}$$

From the equations in (2.14), the multiplier ϕ is 0: When $\phi > 0$, L has to be zero to guarantee the constraint $L \cdot \phi = 0$. Then, $2 \sum_{k=1}^K (S_k + B_k) + \phi$ has to be greater than zero since S_k and B_k are non-negative values for all k . However, it conflicts with the KKT condition. When ϕ is 0, the following equation can be obtained:

$$L = \frac{1}{K} \sum_{k=1}^K (S_k + B_k), \tag{2.15}$$

which is similar to the KKT condition result in (2.2). Therefore, GWF could be the optimization tool for the basic load allocation problem. The reference level L can be regarded as the water level in water-filling concept, and P_T represents the water volume in the water tank. The inelastic load power over time slots is considered as steps inside the water tank. Fig. 2.4 illustrates the power allocation scheme for the 2-D geometric water-filling problem.

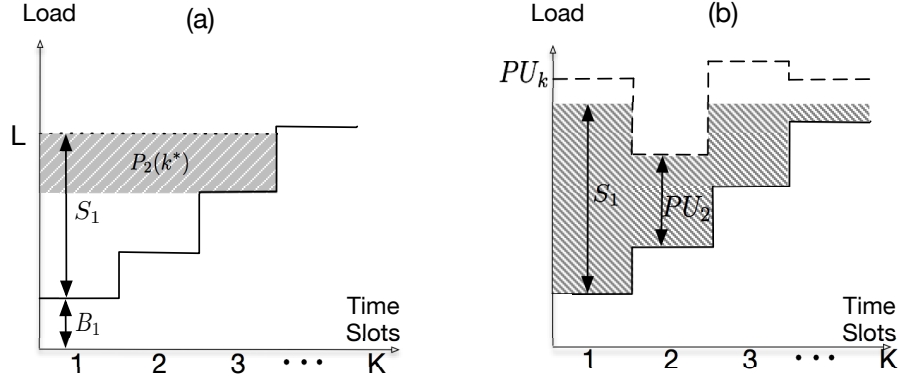


Figure 2.4: Illustration of the basic load allocation scheme.

Fig. 2.4(a) shows the water-filling approach with infinity PU_k , *i.e.*, no upper bound power constraint at each time slot. The steps or inelastic load power B_k are sorted into a monotonic increasing sequence. Moreover, the step width or the weight of the power allocation is identical for all the time slots. The GWF solution, which refers from (2.3) to (2.6), for the elastic load power S_k is shown following:

$$S_k = \begin{cases} S_{k^*} + (B_{k^*} - B_k), & 1 \leq k \leq k^* \\ 0, & k^* < k \leq K, \end{cases} \quad (2.16)$$

where the highest step below the reference level L is

$$k^* = \max\{k | P_2(k) > 0, 1 \leq k \leq K\}, \quad (2.17)$$

and the elastic load power allocated in the k^* th step is

$$S_{k^*} = \frac{1}{k^*} P_2(k^*). \quad (2.18)$$

The water volume above the step k is defined as

$$P_2(k) = \left\{ P_T - \left[\sum_{l=1}^{k-1} (B_k - B_l) \right] \right\}^+, \quad k = 1, \dots, K. \quad (2.19)$$

After the power allocation is solved, the water level L is solved as:

$$L = \frac{1}{k^*} \sum_{k=1}^{k^*} (B_k + S_k) = \frac{1}{k^*} \left(\sum_{k=1}^{k^*} B_k + P_T \right). \quad (2.20)$$

It is assumed that the elastic load power is sufficient enough for the load balance purpose, such that $k^* = K$.

Moreover, Fig. 2.4(b) shows the impact of the upper bound constraint, PU_k , which is indicated by dashed line in the figure. The allocated elastic load power S_k , which is indicated by shadowed areas, can not be higher than the constraint PU_k in time slot k . Algorithm 1 is proposed to solve the problem. Firstly, the set \mathbf{E} includes the indexes of time slots where the elastic load power has not been allocated. And the equations for GWF approach (2.16)-(2.20) are solved without considering PU_k . Then set Λ stores all the indexes of times slots where the allocated elastic load power violates the corresponding constraint. Lines 5 to 8 present that the elastic load power is allocated by the upper bound value, PU_k , for each time slot k in the set Λ , and the set Λ is removed from \mathbf{E} . The algorithm will repeat until no more time slot needed to be assigned, *i.e.* $\mathbf{E} = \emptyset$. By Algorithm 1, the 2-D problem shows in Fig. 2.4 can be solved with the finite computation time and satisfying the upper bound constraint. Therefore, the solution can be simplified as a mapping function with the inelastic load power \mathbf{B} , the upper bound constraint \mathbf{PU} and the total elastic power budget

P_T as:

$$(\mathbf{S}, L) = BLA(\mathbf{B}, \mathbf{PU}, P_T). \quad (2.21)$$

Algorithm 1 Basic Load Allocation (BLA) Water-filling Problem Algorithm

Input: $\mathbf{B}, \mathbf{PU}, P_T, \mathbf{E} = \{1, \dots, K\}$

Output: \mathbf{S}, L

```

1: while  $\mathbf{E} \neq \emptyset$  do
2:   Solve equations (2.16)-(2.20) to obtain  $\{\mathbf{S}\}_{k \in \mathbf{E}}$  and  $L$  via  $\{\mathbf{B}\}_{k \in \mathbf{E}}$  and  $P_T$ .
3:    $\Lambda \leftarrow \{k \mid S_k > PU_k, k \in \mathbf{E}\}$ .
4:   if  $\Lambda \neq \emptyset$  then
5:     if  $k \in \Lambda$  then
6:        $S_k = PU_k$ .
7:     end if
8:      $\mathbf{E} \leftarrow \mathbf{E} \setminus \Lambda, P_T = P_T - \sum_{k \in \Lambda} PU_k$ .
9:   else
10:    Set  $\{S_k\}$  when  $k \in \mathbf{E}$ .
11:     $\mathbf{E} = \emptyset$ .
12:   end if
13: end while

```

2.4 Offline Elastic Load Power Allocation

With full prediction of users' inelastic load power matrix \mathbf{A} , the elastic load allocation \mathbf{R} can be obtained by Algorithm 2. The *BLA* algorithm is called by three times to achieve the 3-dimensional elastic load allocation. The procedures are illustrated graphically in Fig. 2.5. By the first calling *BLA*, the elastic load power allocation for the grid \mathbf{S} among the time slots and the reference level L are determined as shown in Fig. 2.5(a). The shadowed areas indicates the overall elastic load budget, and $\sum_{k=1}^K S_k = P_T$. \mathbf{PU} is the upper bound constraint for the whole grid stage. And the reference level L can be found as follows:

$$L = \frac{1}{K} \left(\sum_{k,j} A_{k,j} + P_T \right) \quad (2.22)$$

Afterwards, Fig. 2.5(b) illustrates Line 3 of Algorithm 2 in power vs. groups graph. The power budget in this stage is the overall elastic load in a time slot, such as S_1 in the figure. The step height represents the overall inelastic load power in the group in one time slot, $\{\sum_{j \in \chi_i} A_{k,j}\}_{i=1}^I$. By calling the *BLA* the second time, the elastic load power in group i and time slot k , $R_{k,i}$ or $\sum_{j \in \chi_i} R_{k,j}$, is allocated with the constraint \mathbf{PG} . Finally, shown in Fig. 2.5(c), the elastic load allocation for the individual user $R_{k,j}$ is determined by *BLA* function in the dimension of power and users in a group, where the power budget is $R_{k,i}$ obtained in the previous stage. The user constraint \mathbf{P} is considered in Line 4 as well. After all, the elastic load power for all the users and time slots is processed by offline ELPA algorithm with guaranteeing the peak power constraints being met by these three stages. Meanwhile, the overall power allocation can be flattened maximally.

Algorithm 2 Offline Elastic Load Power Allocation (Offline ELPA)

Input: \mathbf{A} , \mathbf{PU} , \mathbf{PG} , \mathbf{P} , P_T , $j \in \{1, \dots, J\}$, $i \in \{1, \dots, I\}$, $k \in \{1, \dots, K\}$

Output: \mathbf{R} , reference level L

- 1: $\mathbf{B} = \sum_{i=1}^I \sum_{j \in \chi_i} A_{k,j}$. The vector \mathbf{B} works as parameter vector and it is not changed in this algorithm.
 - 2: $(\mathbf{S}, L) = \text{BLA}(\mathbf{B}, \mathbf{PU}, P_T)$. Steps of BLA are in **Algorithm 1**.
 - 3: $\{R_{k,i}\}_{i=1}^I = \text{BLA}(\{\sum_{j \in \chi_i} A_{k,j}\}_{i=1}^I, \{PG_{k,i}\}_{i=1}^I, S_K)$.
 - 4: $\{R_{k,j}\}_{j \in \chi_i} = \text{BLA}(\{A_{k,j}\}_{j \in \chi_i}, \{P_{k,j}\}_{j \in \chi_i}, R_{k,i})$.
 - 5: Move to next time period, $k^{(2)} = \{1, \dots, K^{(2)}\}$, and back to step 1.
-

2.5 Performance Evaluation

In this section, the performance of offline ELPA algorithm is evaluated and compared with the result from the benchmark work [5]. The load information for household appliances is predicted before the simulation runs. The simulation parameter setting for a household is the same as the setting in [5] shown in Table 2.2. Assume that there are four communities or groups to be evaluated, and in each community, 100 users are participating the load balancing algorithm. It is assumed that elastic loads have full elasticity on the starting time

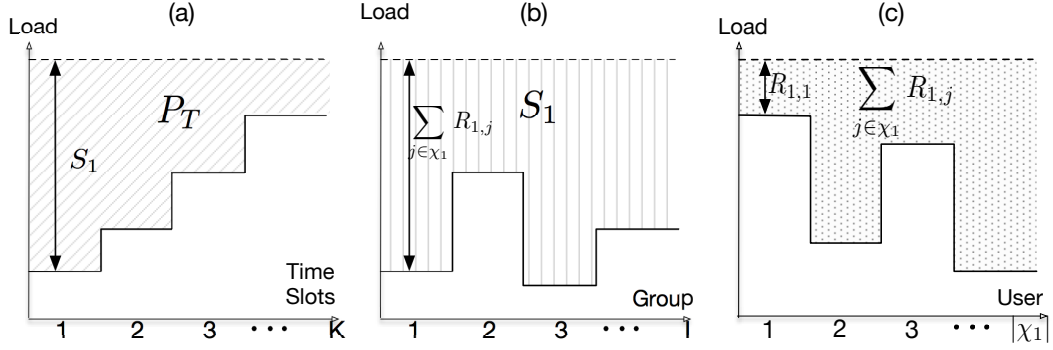


Figure 2.5: (a) Elastic load allocation among time slots. (b) Elastic load allocation among groups. (c) Elastic load allocation among users in a group.

in a day. Once the elastic load is scheduled, the load will operate continuously without being interrupted. The length of one time slot is 30 minutes, and the elastic loads are allocated in 3 days. All the demand information is available from the prediction in a day ahead.

Table 2.2: Household Appliances Parameter Setting

Appliance	Start Time	Operation Time	Soft Load
Cloth Dryer	Normal Dist. Mean at 5PM	1 Hour	Yes
Electric Vehicle	Normal Dist. Mean at 6PM	5 Hours	Yes
Cloth Washer	Normal Dist. Mean at 5PM	0.5 Hour	Yes
Climate Control	Uniform Dist. Over 24 hours	5 Hours	No
Water Heater	Uniform Dist. From 8AM to 12AM	6 Hours	No
Range (1st run)	Normal Dist. Mean at 1PM	1 Hour	No
Range (2nd run)	Normal Dist. Mean at 6PM	1 Hour	No
Electronics	Uniform Dist. From 3PM to 1AM	5 Hour	No
Lighting	Uniform Dist. From 8AM to 1AM	6 Hour	No
Fridge	Uniform Dist. Over 24 hours	24 Hour	No
Kitchen App. (1st run)	Normal Dist. Mean at 1PM	1 Hour	No
Kitchen App. (2nd run)	Normal Dist. Mean at 6PM	1 Hour	No

Fig. 2.6 depicts the overall power demand by applying the offline ELPA algorithm in 3 days. The inelastic component, shown as the dotted curve, conducts as a parameter which can not be scheduled. The elastic loads, shown as the dashed curve, fill the valleys of the inelastic load power in the non-peak time. Meanwhile, when the inelastic load power is increasing, the elastic load power is shaped to decrease correspondingly. The summation of

the elastic and inelastic load power, as the solid curve in the top of the figure, shows less power fluctuation.

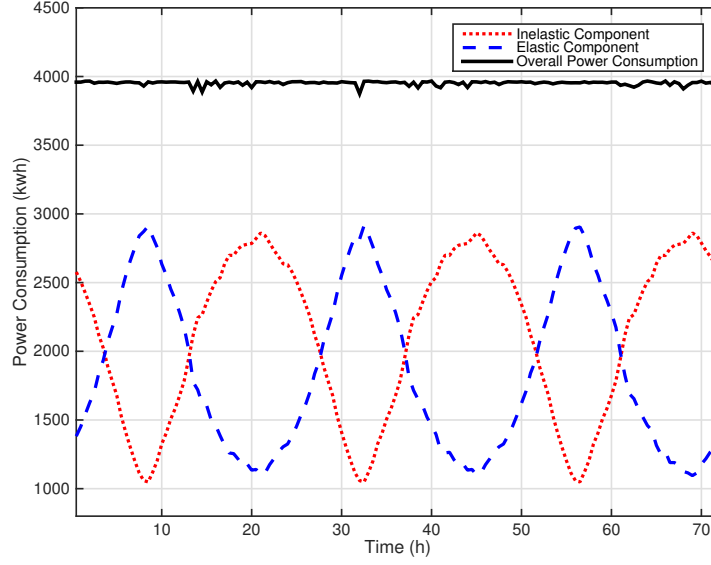


Figure 2.6: Power consumption of inelastic, elastic, and total loads scheduled by offline ELPA for three days.

A comparison with the power demand by the scheduling scheme proposed in [5], referred as “SDWF”, and unscheduled power demand shows in Fig. 2.7. The overall power demand for the three schemes is evaluated in the figure. The ELPA algorithm, showing as solid curve, presents the best performance on flattening the power demand; while SDWF scheme still significantly reduces the overall demand fluctuating compared with the unscheduled case. The standard deviations and the peak to mean load ratios are analyzed for the three schemes in Table 2.3. In the range of the simulation, the overall power demand for offline ELPA reduces the fluctuation with the lowest standard deviation value (14.9) and peak to average ratio (1.004), while the unscheduled case has much higher standard deviation value (1650) and the peak-to-average ratio (1.681). SDWF (standard deviation: 268.4, peak to average ratio 1.105) still flattens the demand compared with the unscheduled case.

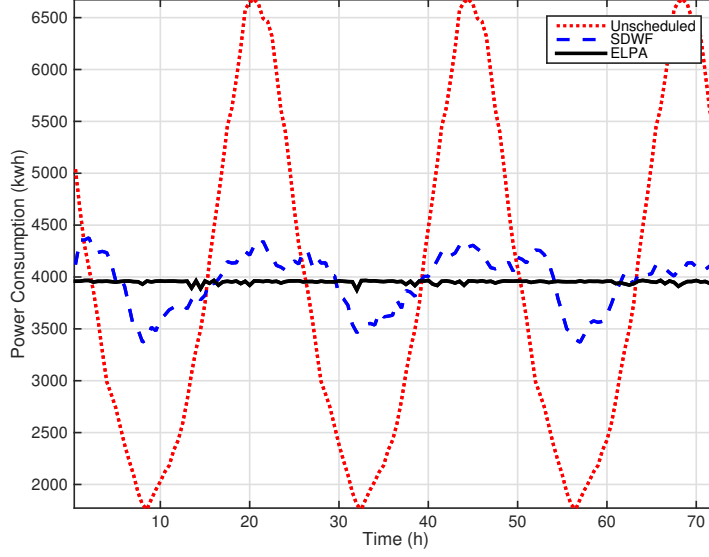


Figure 2.7: Free-run (Unscheduled) total power consumption and the power consumption achieved by Offline ELPA and SDWF for three consecutive days.

Table 2.3: Simulation Results for Offline ELPA, SDWF, and Unscheduled Case

	ELPA	SDWF	Unscheduled
Standard Deviation	14.9	268.4	1650.0
Peak-to-mean Ratio	1.004	1.105	1.681

Figs. 2.8 and 2.9 show elastic load power allocation with the upper bound constraints \mathbf{PU} and \mathbf{PG} respectively. In Fig. 2.8, \mathbf{PU} is set to be 2000 kWh for all the time slots. The elastic load power is bounded in 2000 kWh in the figure. Meanwhile, the reference level is increased correspondingly to adopt all the power budget in the three days. Furthermore, Fig. 2.9 shows the performance comparison between the allocated power demand without \mathbf{PG} and the allocated power demand with \mathbf{PG} . Figs. 2.9(a) and 2.9(c) depict the elastic load power in the group view within 5 hours. Figs. 2.9(b) and 2.9(d) show the overall power demand profile for a day. The group upper bound \mathbf{PG} in Group 4 is set to be 600 kWh for the whole day in Figs. 2.9(c) and (d), while power constraints are set to be large enough in Figs. 2.9(a) and (b). From the figure, the elastic load power in Group 4 is bounded

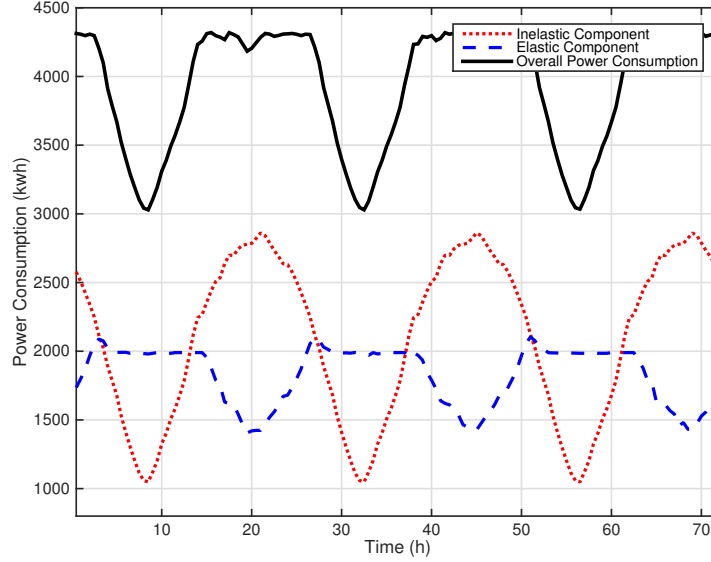


Figure 2.8: Power consumption of inelastic, elastic, and total loads, where PU is 2000 kWh.

under 600 kWh, and the elastic load in other groups is adjusted correspondingly to reach the overall reference level in Fig. 2.9(c). The overall load balancing with the finite upper bound \mathbf{PG} has not been influenced comparing with the case without \mathbf{PG} .

2.6 Chapter Summary

In this chapter, offline load balancing for the smart grid is introduced with the centralized manner. The water-filling approach is applied as a fundamental concept for the proposed load balancing solution. Then, a three-dimension power allocation model is presented in the chapter, which includes the tiers of time slots, groups, and users. To simplify the proposed model, the 3-D problem is split to a few 2-D power allocation problems in the three layers, and GWF approach is utilized as the optimization tool to shape the elastic load power for each layer. The basic load allocation (BLA) projection is proposed to deal with the 2-D problem, and offline elastic load power allocation (ELPA) algorithm is developed from BLA to allocate

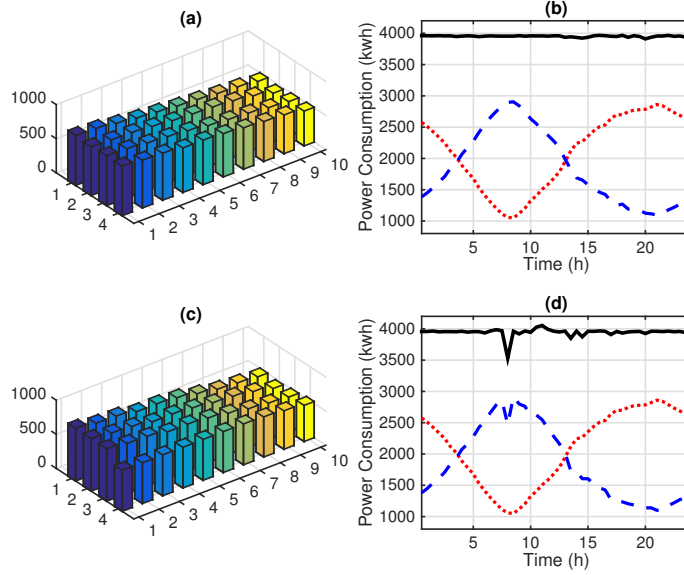


Figure 2.9: (a) Elastic load power consumption in group view, where PG is infinity in five hours, 30 minutes in a time slot. (b) The power consumption of inelastic, elastic, and total loads, where PG is infinity in a day. Legend is the same as Fig. 2.6. (c) Elastic load power consumption in group view, where Group 4 PG is bounded in 600 kWh in five hours, 30 minutes in a time slot. (d) The power consumption of inelastic, elastic, and total loads, where Group 4 PG is bounded in 600 kWh in a day. Legend is the same as Fig. 2.6.

power in the 3-D model. Simulation results show the improvement regarding the fluctuation reduction under the appropriate power constraints as expected, and the optimality of the proposed solution is proofed. In this chapter, the water-filling approach, as a well-known optimization tool for the wireless communication network, is explored in the area of smart grid. The objective for load balancing problem is established as the fundamental work for further discussion.

Chapter 3

Dynamic Centralized Load Balancing Implementation

According to the framework of the offline load balancing scheme, a real-time centralized elastic load allocation implementation is discussed in this chapter. The future load information is no longer available for load balancing algorithm. Two fundamental problems are highlighted in the online scheme: how to define the reference level to accommodate with current load budget, and how to organize the elastic load to balance the loads without losing user's satisfaction. In this chapter, the online network structure is presented. Two algorithms are proposed in the following: Online Elastic Load Power Allocation (OELPA) scheme and Computation Efficient Online Elastic Load Allocation (EOELPA) approach. Furthermore, related simulation results are presented to verify the performance of the proposed algorithms.

3.1 System Model for Online Approach

3.1.1 Network Structure

Unlike the offline scenario which requires one-way elastic allocation from the smart grid operator to the users, the elastic load allocation in real-time needs full communication among smart grid operator, groups, and users. Fig. 3.1 presents the power distribution and communication model for the online centralized load balancing problem. There are two parts in the centralized load balancing network: the electric connection and the communication connection. The electric connection denotes the power line to distribute the power from the energy distributor to the users. The communication network is connected to all nodes in each level in order to control the power demand. Multiple communication protocols can be utilized to establish the centralized load management, such as power line communication and Internet of Things (IoT).

The same as the offline case, there are three levels in the system: the energy distributor, groups, and users. Corresponding constraints are applied to protect transformers in each level. Unlike the offline case, the reference level is dynamically adjusted according to the elastic load budget and power history in real-time. The reference level is a vector \mathbf{L} regarding the time in the online approach, rather than a fixed number in the offline case. To establish the online reference level adjustment, firstly, the smart grid operator evaluates the reference level for the time slot k , L_k , from the load profile of previous N time slots. Fig. 3.2(a) shows the moving window model for the online approach. In the current time slot k , the reference level L_{k+1} is determined by the demand history for time slots from $k - (N - 1)$ to k . The window length N influences the load balancing performance as follows: If the length is small, the reference level tends to stay on the unscheduled power profile. Otherwise, with a large number of N , the reference level fluctuation will be smaller, but the real-time power budget may not be accommodated. After the reference level L_k is defined, the smart grid operator computes the strategy to balance the load and guides the energy distributor to

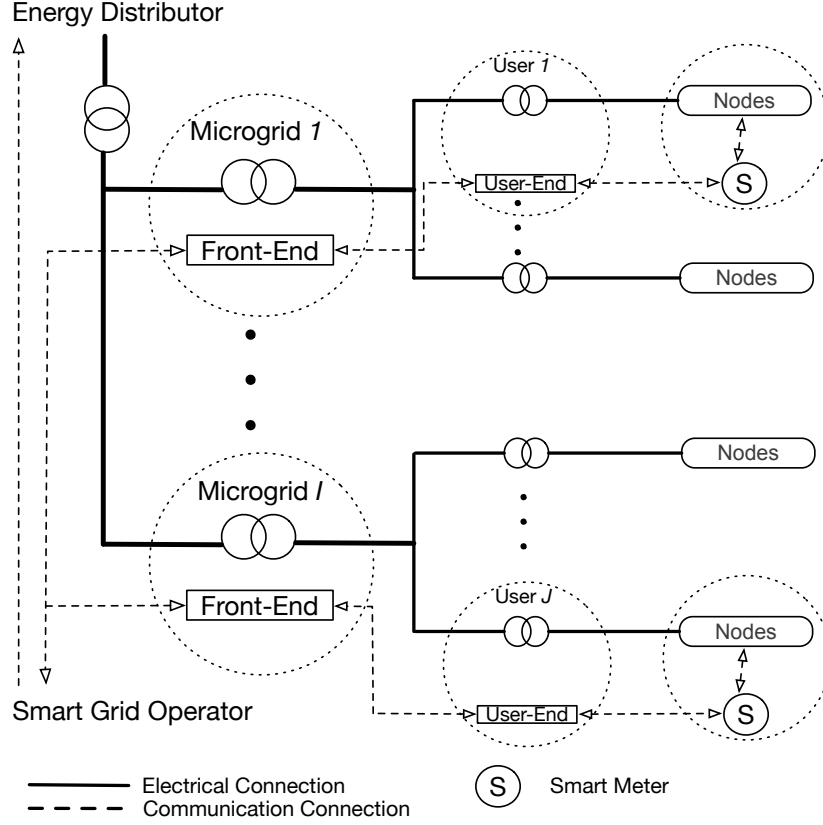


Figure 3.1: Network model for the centralized load balancing problem.

distribute the energy by corresponding load balancing result. In this step, the upper bound PU_k is considered to protect the transformer for the whole system. Furthermore, the smart grid operator balances the elastic load among groups with constraint \mathbf{PG} according to the reference level L_k . Then the front-end of the group allocates the elastic load power for users in its group with the bound \mathbf{P} . Meanwhile, the updated load information is gathered from smart meters in the user ends, and the information will be shared with groups and the smart grid operator to evaluate the reference level for next cycle referring to the power profile history and the real demand pattern.

With the network structure proposed in the previous paragraph, the communication cycle and load balancing algorithm will be applied in each time slots. Centralized demand

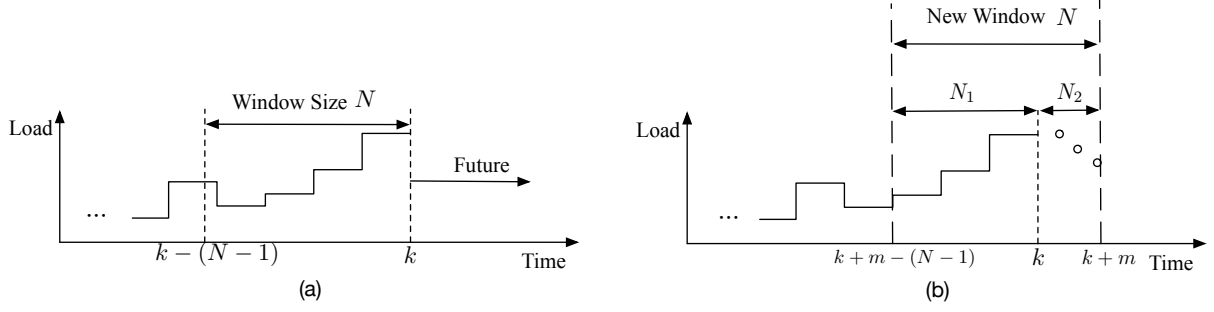


Figure 3.2: Moving Window Model

management in a large-scale power grid will bring the heavy burden of the computation and communication in the system. The computation efficient online algorithm with lower computational complexity is implemented to reduce the computational complexity in the real-time power allocation. To reduce the computation, elastic load power for a period is allocated once. Fig. 3.2(b) shows the moving window model for determining the reference level in the computation efficient online algorithm. The inelastic load power for future m slots are predicted from the window of $[k - N + 1, k]$. Then, the reference level for m time slots is predicted once by the previous power history from $k + m - (N - 1)$ to k . After the reference level for m slots is determined, the new moving window is established to solve the load balancing solution for future m slots, which contains the past window N_1 and future window N_2 . The window N_1 is the slots from $k + m - (N - 1)$ to k , and N_2 is $k + 1$ to $k + m$. The power allocation in m slots will be analyzed once. After m time slots elapsed, the load information in windows N_1 and N_2 is evaluated to predict the inelastic power for next m time slots.

3.1.2 Problem Formulation

The objective function for the dynamic load balancing approach is presented as follows:

$$\begin{aligned}
\min_{\mathbf{R}} \quad & \sum_{k \in N} \left[\sum_{i=1}^I \sum_{j \in \chi_i} (A_{k,j} + R_{k,j}) - L_k \right]^2 \\
\text{subject to} \quad & 0 \leq R_{k,j} \leq P_{k,j}, \quad j = 1, \dots, J, \quad \forall k; \\
& \sum_{j \in \chi_i} R_{k,j} \leq PG_{k,i}, \quad \forall i, k; \\
& \sum_{j=1}^J R_{k,j} \leq PU_k, \quad \forall k; \\
& P_{Smin} \leq \sum_{k=1}^K \sum_{j=1}^J R_{k,j} \leq \sum_{k=1}^K P_{Sk}.
\end{aligned} \tag{3.1}$$

The same as the objective function (2.8) of the offline case, the fluctuation is minimized by reducing the deviation of the overall demand from the reference level. However, different from the offline case, the reference level is a time-varying variable. The term P_{Sk} is the overall elastic load budget in the k th time slot. The scheduled elastic load power should not over than the overall elastic budget in the past slots. The lower bound P_{Smin} is the minimum elastic load power to be scheduled in the period from 1 to K . In this study, it is assumed that the tolerance for the elastic load delay is high enough, *i.e.*, $P_{Smin} = 0$.

3.2 Online Elastic Load Power Allocation

To optimize the objective function (3.1), the reference level should be determined at first. Algorithm 3 is proposed to predict the reference level for the next time slot. Acc_k is the elastic load power budget which is not scheduled yet. The variable Δ is determined to evaluate the difference between the current reference level and the overall load power in the k th time slot. The reference level adjustment procedures are shown in Line 3 to Line 9. The two parameters, ϵ and ξ , are the tolerance for the variables Δ and Acc_k respectively, which are determined by the smart grid operator. If Acc_k or Δ can not satisfy the tolerance,

the new reference level will be defined by getting the average of power demand in the past time window N and the current elastic power budget Acc_k , which is shown in Lines 4 to 6. Otherwise, the reference level will keep as the previous reference level. With small tolerance ϵ and ξ , the reference level will frequently be adjusted. While with large tolerance ϵ and ξ , the reference level adjustment steps will be skipped, while the performance will be influenced since the reference level may not accommodate the current power budget.

Algorithm 3 Reference Level Adjustment

Input: Inelastic and elastic load consumption $A_{n,j}$ and $R_{n,j}$ where $n \in [k-N+1, k]$, L_k , and Acc_{k-1} .

Output: L_{k+1} .

- 1: $Acc_k = Acc_{k-1} + P_{S_k} - \sum_{i=1}^I \sum_{j \in \chi_i} R_{k,j}$.
 - 2: $\Delta = [L_k - (\sum_{i=1}^I \sum_{j \in \chi_i} R_{k,j} + \sum_{i=1}^I \sum_{j \in \chi_i} A_{k,j})]^2$.
 - 3: **if** $\Delta \geq \epsilon$ **or** $Acc_k \geq \xi$ **then**
 - 4: $P_T = \sum_{n=k-N+1}^k \sum_{i=1}^I \sum_{j \in \chi_i} R_{n,j} + Acc_k$.
 - 5: $B = \sum_{n=k-N+1}^k \sum_{i=1}^I \sum_{j \in \chi_i} A_{n,j}$.
 - 6: $L_{k+1} = \frac{1}{N} (P_T + B)$.
 - 7: **else**
 - 8: $L_{k+1} = L_k$.
 - 9: **end if**
-

With the reference level adjustment algorithm, the online elastic power load allocation (OELPA) algorithm is proposed in Algorithm 4. The same as Algorithm 2, there are three levels elastic power allocation in OELPA: the entire grid, groups, and users. However, the elastic load power for the grid, S_k , is obtained from the reference level given by the previous loop and the inelastic load for the system in the k th time slot, which is shown in Line 1 of Algorithm 4. The constraint PU_k will substitute S_k if the elastic load power violates the constraint. Line 5 and Line 6 present the elastic load power allocation at the group level and the user level respectively, which is the same as Lines 3 and 4 in ELPA algorithm. Afterwards, the reference level adjustment algorithm is called for evaluating the reference level for the next time slot. The moving window moves forward by one time slot as well. OELPA algorithm allocates the elastic load power in each time slot. When the future load

information is unknown or predicted unreliably, the algorithm offers an online solution to schedule the elastic load in the real-time with the adjustable reference level, while the offline ELPA can not achieve the objective for the fixed reference level.

Algorithm 4 Online Elastic Load Allocation Algorithm (OELPA)

Input: Inelastic load consumption $A_{k,j}$, PU_k , $PG_{k,i}$, and $P_{k,j}$ at time k , where user $j = 1, \dots, J$, group $i = 1, \dots, I$, and window size N .

Output: Elastic load allocation at time k , $R_{k,j}$.

- 1: $S_k = L_k - \sum_{j=1}^J A_{k,j}$, where L_k is obtained from the previous loop.
 - 2: **if** $S_k \geq PU_K$ **then**
 - 3: $S_k = PU_K$.
 - 4: **end if**
 - 5: $\{R_i\}_{i=1}^I = \text{BLA}(\{\sum_{j \in \chi_i} A_{k,j}\}_{i=1}^I, \{PG_{k,i}\}_{i=1}^I, S_k)$.
 - 6: $\{R_{k,j}\}_{j \in \chi_i} = \text{BLA}(\{A_{k,j}\}_{j \in \chi_i}, \{P_{k,j}\}_{j \in \chi_i}, R_i)$.
 - 7: Run reference level adjustment algorithm.
 - 8: Move to next time slot $k+1$. Back to step 1.
-

3.3 Computation Efficient Online Approach

In OELPA algorithm, load balancing algorithm is applied for every time slot with small tolerance, ϵ and ξ . However, to ensure the reference level accommodating the real-time grid information, ϵ and ξ should be set as small as possible, which increases the volume of computation in the large-scale system. Then the computation efficient online elastic load allocation (EOELPA) approach is proposed in Algorithm 5. The computation is reduced by generating the elastic load allocation for future multiple time slots. The computation for the central controller could be reduced by less load allocation process. Compared with OELPA, EOELPA allows that the central controller communicates with other nodes and generates elastic load allocation once every m time slots.

Line 1 to Line 3 shows the inelastic load prediction process. Referring to the moving window model in Fig. 3.2(b), first, the inelastic load power in the past window, $A_{k-N+1,j}$ to $A_{k,j}$, is fitted into the cubic spline to predict the inelastic load power, $\hat{A}_{k+1,j}$ to $\hat{A}_{k+m,j}$,

Algorithm 5 Computation Efficient Online Elastic Load Allocation Algorithm (EOELPA)

Input: Inelastic load consumption \mathbf{A} in the past window $[k - N + 1, k]$, \mathbf{PU} , \mathbf{PG} , and \mathbf{P} at future m slots, for all users $j = 1, \dots, J$, groups $i = 1, \dots, I$.

Output: Elastic load allocation \mathbf{R} for future m time slots.

- 1: Fit inelastic load consumption $A_{n,j}$ to cubic spline, where $n \in [k - N + 1, k]$.
 - 2: Predict inelastic load consumption $\hat{A}_{n_2,j}$ by previous spline, where $n_2 \in [k + 1, k + m]$.
 - 3: Window moves from $[k - N + 1: k]$ to $[k + m - N + 1: k + m]$, where N_1 is the past part in the window, N_2 is the future part in the window. $N_1 = [k + m - N + 1: k]$, $N_2 = [k + 1: k + m]$.
 - 4: Compute L_{N_2} at time k by Algorithm 6.
 - 5: $S_{N_2} = L_{N_2} - \sum_{j=1}^J \hat{A}_{N_2,j}$.
 - 6: **if** $S_{N_2} \geq PU_{N_2}$ **then**
 - 7: $S_{N_2} = PU_{N_2}$.
 - 8: **end if**
 - 9: $\{R_i\}_{i=1}^I = \text{BLA}(\{\sum_{j \in \chi_i} \hat{A}_{n_2,j}\}_{i=1}^I, \{PG_{n_2,i}\}_{i=1}^I, S_{N_2})$.
 - 10: $\{R_{n_2,j}\}_{j \in \chi_i} = \text{BLA}(\{\hat{A}_{n_2,j}\}_{j \in \chi_i}, \{P_{n_2,j}\}_{j \in \chi_i}, R_i)$.
 - 11: Back to step 1 after m time slots are elapsed.
-

for the future window N_2 . The parameter, m , denotes the future window length influencing the prediction accuracy and the computation load. With the longer future window length, the computation load is reduced, but the accuracy is decreased for the uncertain prediction. Otherwise, the prediction accuracy is increased with the shorter future window length, but the computation load is close to OELPA approach. After the future information is predicted, the window will move forward by m slots, where N_1 is the time slots set for the past window, and N_2 is the time slots set for the future window.

Line 4 calls the corresponding reference level prediction algorithm proposed in Algorithm 6. With the same scheme of Algorithm 3, the reference level is predicted by the load information history and the elastic load power budget which has not been satisfied yet. However, the reference level for multiple slots is predicted once, while Algorithm 3 only predicts reference level for a time slot. Afterwards, the elastic load for future m time slots can be scheduled by the water-filling approach in three levels from Lines 5 to 10. After m time slots elapsed, the moving window will move forward for the next m slots to proceed the allocation repeatedly.

Algorithm 6 Reference Level Prediction Algorithm

Input: Inelastic and elastic load consumption $A_{n_1,j}$ and $R_{n_2,j}$ where $n_1 \in N_1$, predicted inelastic load consumption $\hat{A}_{n_2,j}$ where $n_2 \in N_2$, the elastic load demand P_S in past m slot and Acc_{k-m} .

Output: L_{N_2} .

- 1: $Acc_k = Acc_{k-m} + \sum_{n=k-m+1}^k P_{Sn} - \sum_{n=k-m+1}^k \sum_{j=1}^J R_{n,j}$.
 - 2: $P_T = \sum_{n_1 \in N_1} \sum_{j=1}^J R_{n_1,j} + Acc_k$.
 - 3: $L_{N_2} = \frac{1}{N} (P_T + \sum_{n_1 \in N_1} \sum_{j=1}^J A_{n_1,j} + \sum_{n_2 \in N_2} \sum_{j=1}^J \hat{A}_{n_2,j})$.
-

Compared with the OELPA algorithm, EOELPA algorithm reduces the computation burden by $1/m$ of the computation load in OELPA since the elastic power is allocated once for every m time slots in EOELPA. The communication network cycle in Fig. 3.1 only exchanges the information once for m time slots with EOELPA algorithm, while every node has to stay online to transmit the information by each time slot in OELPA algorithm. From the comparison above, EOELPA shows the advantage on the reducing computation complexity. However, in the term of the accuracy regarding the prediction and accommodation with the real-time grid information, OELPA is expected to have less power fluctuation than EOELPA due to the uncertainties of the prediction in EOELPA. Therefore, EOELPA is fit for the large scale network or communication limitation network, while OELPA is preferred in the system requiring less fluctuation for the dynamic power analysis.

3.4 Performance Evaluation

In this section, the performance of the online elastic load power allocation scheme (OELPA) and the computation efficient OELPA (EOELPA) is evaluated by comparing the unscheduled power demand. The load model is the same as the offline power allocation, but the future energy demand is unknown. The parameter setting for OELPA algorithm is following: the window size N is 24 hours; the parameters ϵ and ξ are set as 30 kWh and 100 kWh respectively.

Fig. 3.3 illustrates the performance of OELPA algorithm for five consecutive days. The load budget in Fig. 3.3(a) is identical in these five days. The overall power fluctuation in the proposed algorithm is reduced significantly compared with the unscheduled case. As shown in Table 3.1, the standard deviation is reduced from 1603 to 154 by the proposed method. Unlike the offline case, the reference level is a time-varying vector rather than a parameter. In the case with the identical power budget, the reference level is maintained in horizontal. The flattened reference level brings the fluctuation reduction: the standard deviation of the reference level is 57.09, and the overall power demand scheduled by the proposed algorithm follows the reference level closely. The elastic load power fills the demand valley and shaves the demand peak in real-time by adapting the reference level dynamically.

Furthermore, the load budget is adjusted to present dynamic load balancing with different power behaviours. In Fig. 3.3(b), the power budget is increased in five days. The load model is the same as the case with identical power budget, but the probability of operating power loads is growing in the five days. In the unscheduled case, the power budget growth is shown in the increasing peak value. By the proposed OELPA algorithm, the reference level is increased correspondingly to accommodate the accelerating power budget. The reference level is growing from 2742 kWh to 3572 kWh. The overall demand by the proposed algorithm is getting close to the reference level. The power demand is flattened by the proposed algorithm with adopting the increasing power budget. Comparing to the unscheduled power demand, the standard deviation for OELPA is reduced from 1503 to 258 as shown in Table 3.1. The allocation result with decreasing power budget in five days is presented in Fig. 3.3(c). The reference level is descending from 3971 to 3023 kWh to fit the decreasing power budget. The same as Figs. 3.3(a) and (b), the smoothed reference level leads the overall power fluctuation reduction. The power fluctuation is reduced from 1522 to 379 in standard deviation. From the performances shown in Fig. 3.3, the dynamic reference level is able to accommodate the current power budget in real-time without future load information. The moving window provides the solution to generate the dynamic reference level. By the

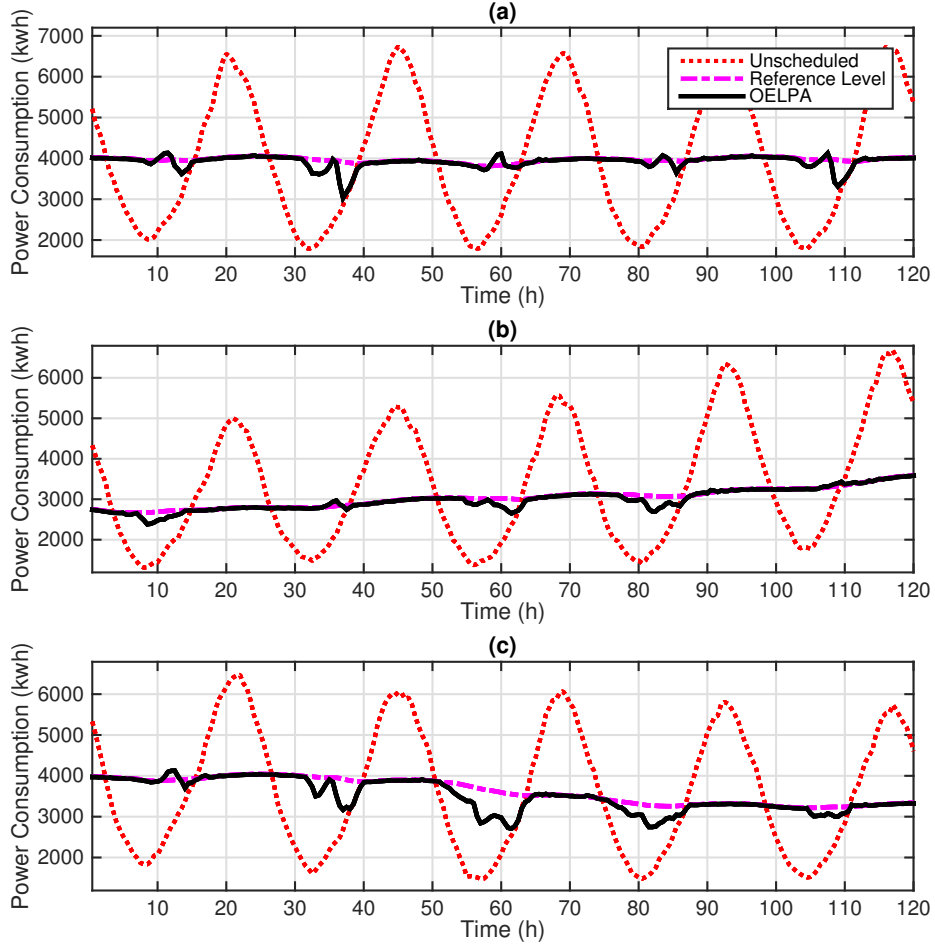


Figure 3.3: Overall power consumption scheduled by OELPA and unscheduled power consumption for 5 days. (a) With same load budget. (b) Load budget increasing. (c) Load budget descending.

predicted reference level, the OELPA algorithm reduces overall power fluctuation in the real-time without the future power demand knowledge.

The performance of the computation reduced online elastic load allocation (EOELPA) algorithm is presented in Fig. 3.4. The overall power demand with EOELPA algorithm is compared with the overall power demand scheduled by OELPA algorithm. Similar to the

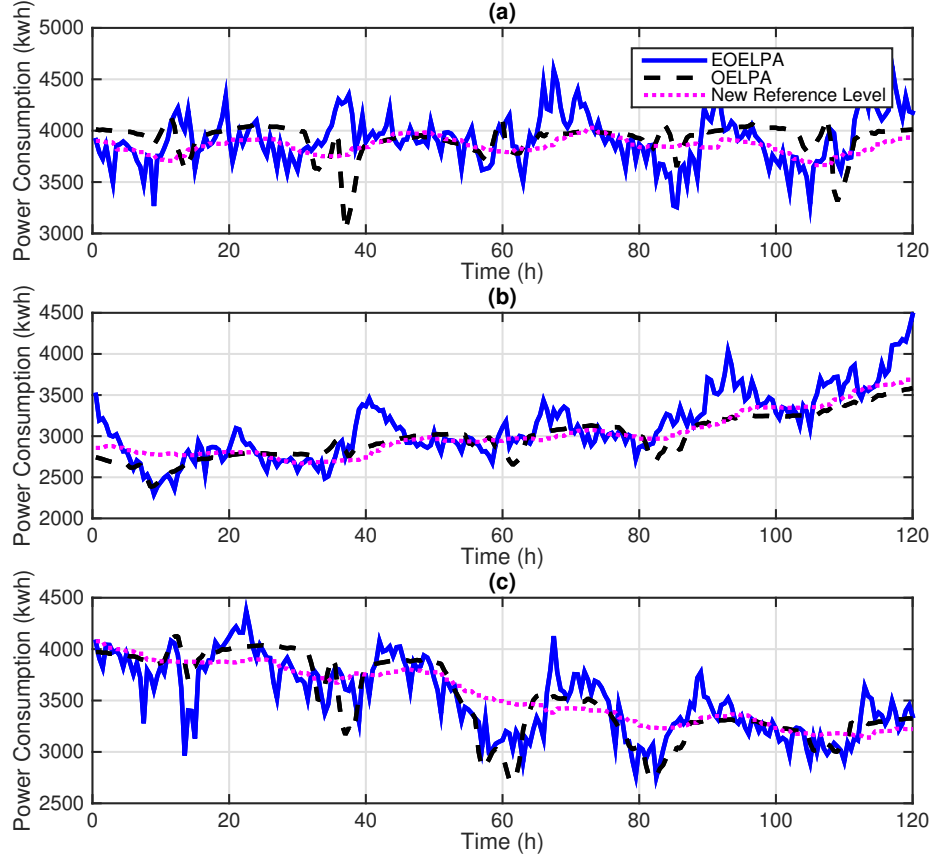


Figure 3.4: Power consumption of overall loads scheduled by EOELPA and OELPA for 5 days. (a) With same load budget. (b) Load budget increasing. (c) Load budget descending.

simulation model in Fig. 3.4, the simulation is run for five days, and three scenarios are considered: the elastic load allocation for the identical power budget, the ascending power budget, and the descending power budget. The inelastic load power in the past window is fit into a set of cubic polynomials. Then the inelastic load demand for the next m time slots is predicted by cubic spline extrapolation. The elastic load power for future m time slots is generated by EOELPA algorithm in every m time slots. In the simulation results shown in Fig. 3.4, the future window length m is 3. The smart grid operator gathers the load information and predicts the reference level for next three time slots in a communication

cycle. Then the smart grid operator stays in offline until the three time slots elapsed. The figure shows that there is more fluctuation in EOELPA results comparing the OELPA results since the prediction accuracy influences the performance of EOELPA while there is no prediction factor in OELPA algorithm. Although the results of EOELPA show more ripples, the demand fluctuation is still reduced significantly compared the unscheduled demand as well. As shown in Table 3.1, the standard deviation of EOELPA is $1/4$ of the results from the unscheduled case. Nevertheless, the reference level adjustment in EOELPA also enables adapting the dynamic demand budget as OELPA, and the computation of EOELPA is reduced to $1/m$ of the computation in OELPA.

Table 3.1: Standard Deviation of Overall Power Consumption in 5 Consecutive Days (Fig. 3.3 and 3.4).

Loads Demand	OELPA	EOELPA	Unscheduled
Identical (a)	154	241	1603
Ascending (b)	258	395	1503
Descending (c)	379	351	1522

From the results in Fig. 3.4, with increasing the parameter m , the computation and the communication burden can be reduced by the longer period for power prediction and allocation. However, the load balancing performance is influenced by the length of the predicted time slots. Fig. 3.5 shows the impact of various m selection to the performance of EOELPA. With m increasing, the standard deviation of the overall power demand allocated by EOELPA is growing correspondingly, while EOELPA still reduces the power fluctuation significantly compared with the unscheduled case. From above simulation results, OELPA can be utilized in the system requiring the smooth power demand since it has better performance on balancing loads. EOELPA is more preferred in the large-scale system with accepting appropriate tolerance on the power fluctuation.

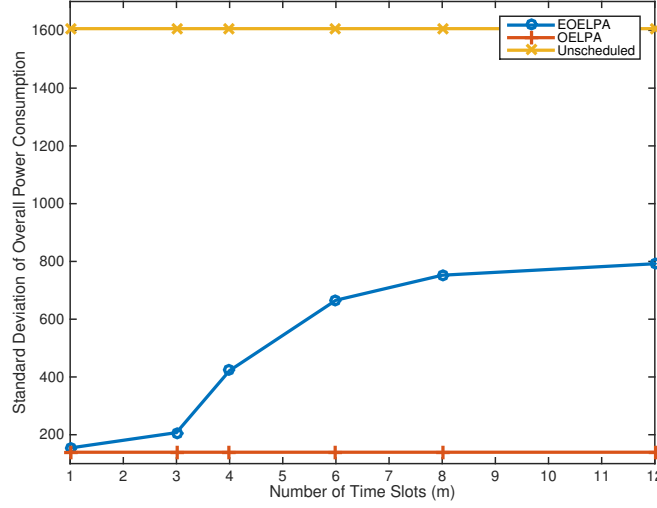


Figure 3.5: Standard deviation vs. parameter m for unscheduled, OELPA, and EOELPA schemes.

3.5 Chapter Summary

This chapter solves the problem of load balancing without the knowledge of future load information. The communication cycle is introduced first for establishing a dynamic load allocation environment. The objective for dynamic load balancing is proposed based on the offline problem. The corresponding power allocation procedures are introduced as online elastic load power allocation (OELPA) algorithm. With OELPA, the elastic load power can be allocated to minimize the power fluctuation in real-time. The reference level can be adjusted by the current power budget and flattened by previous energy information. The power fluctuation is minimized by dynamic reference level adjustment for each time slot. Furthermore, computation efficient online elastic load allocation algorithm (EOELPA) is proposed to reduce the communication and computation burden from OELPA. Unlike OELPA adjusting the reference level in each time slot, EOELPA enables to modify the reference level and allocates elastic load power for future multiple time slots. Corresponding simulation results are investigated. The improvement of the proposed algorithms is presented

by comparing the power demand fluctuation with the unscheduled case. In summary, two effective power allocation schemes are discussed in this chapter when the power prediction is not reliable. OELPA provides the proper elastic load allocation scheme to balance power loads in real-time, and EOELPA developed from ELPA to allocate power demand in the large-scale system with a computation efficient manner.

Chapter 4

Decentralized Load Balancing by EV Fleets

In previous chapters, centralized demand management schemes are discussed to achieve load balancing with the assumption that elastic loads are sufficient enough to fill the valleys and cut the demand peaks for the entire system. However, the elastic power in some distribution sub-networks may be insufficient to balance the loads inside the sub-networks. Fig. 4.1 illustrates the power profile for a resident unit, a retail unit, and an industrial unit respectively, in a day [2]. From the figure, it can be seen that distribution sub-networks with different composition of user types show various electricity load profiles. For example, the demand peak occurs in the evening for some sub-networks where residential units are dominant over other user types, while the power demand is high for the whole day in sub-networks located in central business districts [36]. Therefore, in this chapter, the decentralized power allocation scheme is proposed to balance the power load in the level of sub-networks. Due to mobility and charging elasticity of EVs, EV fleets among neighbouring sub-networks could serve as circulating elastic loads across sub-networks when other elastic loads in some sub-networks are not enough for balancing the load therein. In this chapter, we assume that EV can fleet among neighbouring sub-networks without any power loss. The proposed approach

achieves decentralized load balancing in two steps. Firstly, the optimal load power allocation is calculated via Proximal Jacobian ADMM technique. Two scenarios are discussed in this step: Firstly, the basic decentralized EV power balancing scheme is introduced for the offline power allocation. All EVs are available to be scheduled for charging starting from the first time slot, and the optimal EV power allocation is calculated once for each period with K time slots. Secondly, the dynamic decentralized EV power allocation scheme is proposed as an online solution to find the optimal power allocation in real-time. After the power allocation results are obtained by the offline or the real-time power allocation optimization, a random access scheme is proposed as the second step to schedule the charging of EV units to implement the optimal power allocation result obtained in the first step. Randomness is introduced in the proposed EV charging scheme in order to ensure that: 1) actual charging power to EV units is as close as possible to the optimal EV power allocation result obtained; 2) fairness is achieved among all PHEVs requesting to be charged in the system. Table 4.1 is a list of the variables used in analysis of the decentralized load balancing problem.

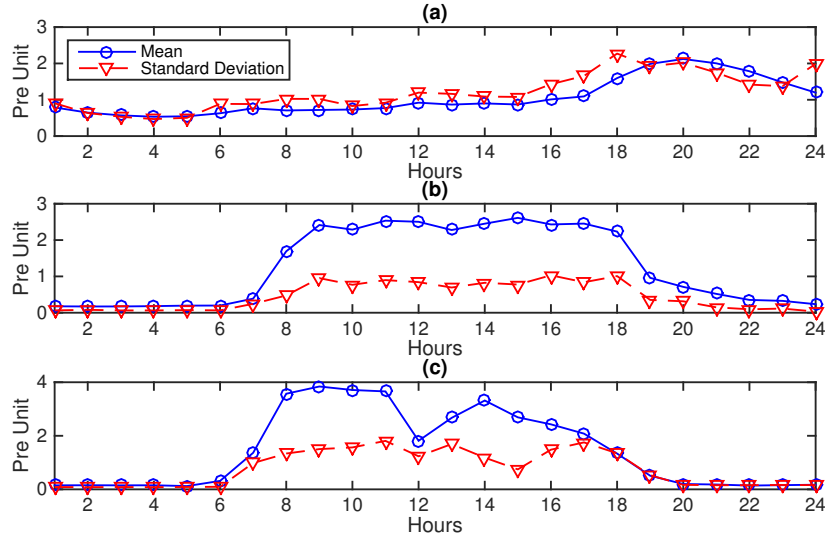


Figure 4.1: Power profile for (a) a residential unit, (b) a retail unit, and (c) an industrial unit in a month with power base $[\text{kWh}/(24 \times 30)]$ [2].

Table 4.1: List of variables for centralized load balancing

Group	Variable	Meaning
Index Group	k	index of time slots, for $k = 1, \dots, K$.
	g	index of charging stations, for $g = 1, \dots, G$.
	e	index of EV units, for $e = 1, \dots, E$.
	χ_g	the set of EVs requesting to be charged in sub-network g for basic EV power allocation.
	$\chi_{g,k}$	the set of EVs requesting to be charged in sub-network g at time k for dynamic EV power allocation.
Load Group	\mathbf{A}	the base load power. The (g,k) th element, $A_{g,k}$, denotes the base load power of sub-network g at time k .
	\mathbf{V}	the power of EVs in the station. The (g,k) th element, $V_{g,k}$, denotes EV load power of sub-network g at time k .
	\mathbf{Q}	the power of newly EVs arriving the station. The (g,k) th element, $Q_{g,k}$, denotes newly arriving EV load power of sub-network g at time k .
	\mathbf{r}	the flowing energy or power by EV fleeing. For basic EV power allocation, $r_{g,h}$ is EV fleeing energy from station g to h . For dynamic EV power allocation, $r_{g,h,k}$ is EV fleeing power from station g to h at time k .
	\mathbf{PO}	EV energy or power shifted out from a station. For basic EV power allocation, PO_g denotes the EV energy shifted out from a station g . For dynamic EV power allocation, $PO_{g,k}$ denotes the EV power shifted out from station g at time k .
	\mathbf{PI}	EV energy or power shifted into a station. For basic EV power allocation, PI_g denotes the EV energy shifted into a station g . For dynamic EV power allocation, $PI_{g,k}$ denotes the EV power shifted into station g at time k .
Constraint Group	\mathbf{PG}	the upper bound of allocated EV power is denoted. The (g,k) th element, $PG_{g,k}$, denotes EV power upper bound for the g th sub-network at time k .
	\mathbf{BI}	the maximum EV energy from other stations. The g th element, BI_g , the maximum EV energy from other stations for station g .
	\mathbf{Vp}	the lower bound for EV power allocation in the charging station. The (g,k) th element, $Vp_{g,k}$, denotes EV power lower bound for the g th sub-network at time k .
EV Parameter	P_c	charging power for an EV.
	Tc_e	the charging time for EV e .
	T_{min_e}	minimum charging time for EV e .
	SOC_e	state of charge for EV e .
	B_e	the battery capacity for EV e .
Reference Level Group	\mathbf{L}	reference level. For basic EV power allocation, L_g denotes the reference level for station g . For dynamic EV power allocation, $L_{g,k}$ denotes the reference value for station g at time k .

4.1 Overview of ADMM

ADMM is a powerful optimization tool for distributed convex optimizations [52–54]. It splits a large global optimization problem into multiple smaller local subproblems. ADMM combines the advantage of decomposition, which enables distributed and parallel optimization, and augmented Lagrangian method, which guarantees convergence under mild conditions [52]. The following problem is introduced to illustrate ADMM:

$$\begin{aligned} \min_{\{x,z\}} \quad & f(x) + g(z) \\ \text{subject to} \quad & ax + bz = c, \end{aligned} \tag{4.1}$$

where f and g are closed proper convex functions. Also, $x \in \mathbf{R}^n$, $z \in \mathbf{R}^m$, $a \in \mathbf{R}^{p \times n}$, and $b \in \mathbf{R}^{p \times m}$. The augmented Lagrangian for the problem (4.1) is shown as the (4.2), where the quadratic penalty of constraint is added on top of the Lagrangian:

$$\mathcal{L}_\rho(x, z, y) = f(x) + g(z) + y^T(ax + bz - c) + (\rho/2)\|ax + bz - c\|_2^2. \tag{4.2}$$

where $\rho > 0$. ADMM finds the solution in an iterative process which treats $f(x)$ and $g(z)$ separately. The multiplier y is updated by the dual ascent method. The update of primal and dual variables in the iterative process is given as follows:

$$x^{k+1} = \operatorname{argmin}_x \mathcal{L}_\rho(x, z^k, y^k), \tag{4.3a}$$

$$z^{k+1} = \operatorname{argmin}_z \mathcal{L}_\rho(x^{k+1}, z, y^k), \tag{4.3b}$$

$$y^{k+1} = y^k + \rho(ax^{k+1} + bz^{k+1} - c). \tag{4.3c}$$

Alternatively, an equivalent scaled form is outlined below. It is a convenient form in which linear and quadratic terms are combined together:

$$x^{k+1} = \operatorname{argmin}_x f(x) + (\rho/2) \|ax + bz^k - c + u^k\|_2^2, \quad (4.4a)$$

$$z^{k+1} = \operatorname{argmin}_z g(z) + (\rho/2) \|ax^{k+1} + bz - c + u^k\|_2^2, \quad (4.4b)$$

$$u^{k+1} = u^k + \rho(ax^{k+1} + bz^{k+1} - c), \quad (4.4c)$$

where $u = y/\rho$.

From the above example, it can be seen that ADMM technique can optimize the problem by optimizing variables x and z , connected through the dual variable y , separately. Therefore, ADMM is a suitable tool for finding the optimal solution of the block-separable objective function in a distributed manner. The conventional ADMM technique shown in the problem (4.1) solves the problem with two blocks, where each block corresponds to a function in the form of f and g . However, the conventional ADMM technique may not converge for the multi-blocks problem when the number of blocks is larger than two unless the objective function for each block satisfies extra conditions such as strong convexity [53, 55]. Proximal Jacobian ADMM technique proposed in [53] splits the multi-blocks problem into multiple subproblems. However, compared with the conventional ADMM technique, the proximal term $\frac{\tau}{2} \|x_i - x_i^k\|^2$ is added in Proximal Jacobian ADMM algorithm to guarantee strict convexity. As a result, it is shown that Proximal Jacobian ADMM method converges under much milder conditions.

The proposed work utilizes Proximal Jacobian ADMM technique to ensure stable convergence since multiple subproblems, one for each charging station, are considered in this work. With Proximal Jacobian ADMM, the convergence rate can be reduced to $o(1/k)$ rather than the convergence rate of $O(1/k)$ in the conventional ADMM [53], where k denotes the iteration number. The detailed procedures are shown using the following optimization problem

with N function blocks as an example:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \sum_{i=1}^N f_i(x_i) \\ \text{subject to} \quad & \sum_{i=1}^N A_i x_i = N \cdot c. \end{aligned} \tag{4.5}$$

To simplify the notation, the constraint in (4.5) can be converted to $\overline{Ax} = c$, where $\overline{Ax} = \frac{1}{N} \sum_{i=1}^N A_i x_i$. Then, by Proximal Jacobian ADMM technique, the global optimization (4.5) can be split into local optimization problems:

$$\begin{cases} x_i^{k+1} = \operatorname{argmin}_{x_i} f_i(x_i) + (\rho/2) \|A_i x_i - c - (A_i x^k - \overline{Ax}^k) + u^k\|_2^2 + \frac{\tau}{2} \|x_i - x_i^k\|^2, \\ \overline{Ax}^{k+1} = \frac{1}{N} \sum_{i=1}^N A_i x_i^{k+1}, \\ u^{k+1} = u^k + (\overline{Ax}^{k+1} - c). \end{cases} \tag{4.6}$$

Using the above solution, each variable x_i can be updated given the current value of x_i and the current value of the mean \overline{Ax} . This facilitates a distributed update of variables. The mean \overline{Ax} is considered as the global information, which converges to c as the local variables coverage to the optimal solution. The algorithm allows parallel computation in optimizing the global objective which is separable and thereby reduces information exchange overhead.

4.2 System Model

Fig. 4.2 shows the grid model for the proposed decentralized load balancing problem. The system is divided into multiple zones in the power grid, each with a bus in its area. In total, there are G neighbouring zones or buses. To simplify the system model, it is assumed that there is one charging station in each sub-network. The charging station is responsible for gathering the local information and communicate with the coordinator for uploading local results and obtaining results from other sub-networks. Unlike the centralized power allocation scheme in Chapter 3, in which a smart grid operator gathers and computes the

load balancing strategy for all users, a coordinator gathers and shares information among the charging stations in all sub-networks in the decentralized scenario. However, the computation of optimal power allocation strategy is performed by each sub-network instead of by the operator. As will be shown in the rest of this chapter, load balancing in the entire system can be achieved by parallel local optimization processes.

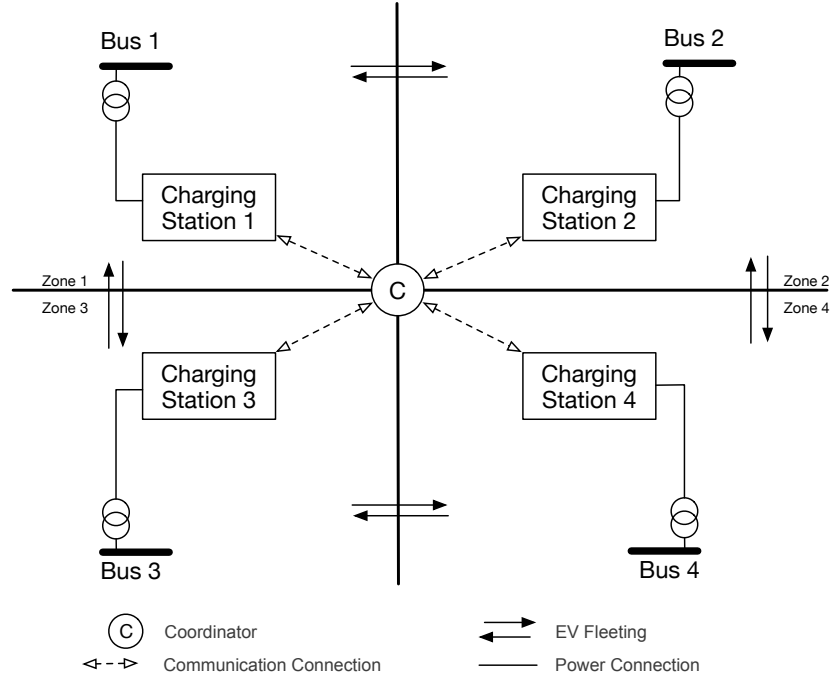


Figure 4.2: System model for EV charging in neighbouring charging stations.

The decentralized load balancing approach is divided into two phases: First, a power allocation optimization phase calculates the optimal EV power for balancing loads among sub-networks. In this phase, the optimal EV power allocated inside the sub-networks and EV power flowing across sub-networks are determined. Second, EV units are scheduled by a randomized charging scheme in order to implement the optimal power allocation derived from the optimization phase. Two scenarios are considered for the first phase, *i.e.*, the basic (offline) scenario and the dynamic (real-time) scenario. It is assumed that EV fleeing will not cause extra cost on EV users and charging stations in either scenario. In the basic

decentralized scenario, all EVs are available to start charging from the first time slot and can expect to be fully charged during the period of $[1, K]$. All base load information is known for power allocation. In contrast, in the dynamic scenario, EVs arrive at charging stations dynamically with different charging time requirements. Load information in the future information is unknown and the charging strategy determined in previous time slots can not be changed.

When an EV arrives at a zone g , the EV sends a charging request to the local charging station. The charging station will determine whether it should accept the request or transfer the request to another charging station. The decision is made based on the power allocation optimization solution. Once the charging station accepts the EV, the EV will start charging and stay at the station until the battery is fully charged. The charging power for an EV is constant over time, which is denoted by P_c . The charging start time and leaving time are parameters defined by the EV users under the assumption that the duration of charging is long enough to fully charge the battery. The time length from current time slot to the slot when EV e leaves is T_{c_e} . After EV e plugged into the system, the charging station will evaluate the EV's state of charge SOC_e . The battery capacity for EV e is denoted as B_e . Then the minimum charging time can be determined by the following equation:

$$T_{min_e} = \left\lceil \frac{(1 - SOC_e)B_e}{P_c} \right\rceil. \quad (4.7)$$

Since the total charging time defined by the user is sufficient to be fully charged, in the first time slot when the EV e plugged in, the rest of charging time T_{c_e} is no less than the minimum charging time T_{min_e} , *i.e.*, $T_{c_e} \geq T_{min_e}$. Comparing the time length between T_{c_e} and T_{min_e} , the charging scheme should consider two different cases. If $T_{c_e} > T_{min_e}$, EV e has flexibility on charging scheduling (*i.e.*, its charging could be interrupted or delayed by at least one time slot). Otherwise, *i.e.*, $T_{c_e} = T_{min_e}$, EV e has to be charged without being interrupted to ensure that it can be fully charged before the vehicle leaves. For the offline

power allocation scheme, the initial power requested to be charged is the same for all EVs. In the following two sections, we first introduce the decentralized EV power allocation for the offline scenario and real-time scenario, respectively, and then propose an EV scheduling scheme for implementing the optimal power allocation obtained in these two scenarios.

4.3 Decentralized EV Power Allocation Optimization Schemes

In this section, two power allocation schemes are introduced to solve the decentralized load balancing problem with EV power being the elastic load. First, a basic decentralized EV power allocation scheme is presented to achieve load balancing in the offline scenario, in which the power allocation is optimized once for every duration with K time slots. The base load profiles for all sub-networks are assumed to be fully known. Second, to accommodate dynamic EV load arrivals and various EVs charging time requirements, a dynamic decentralized EV power allocation scheme is proposed by extending the optimization problem from the basic decentralized EV power allocation methodology. With the proposed dynamic scheme, load balancing among all sub-networks is realized by taking into consideration of the charging requirements in real-time.

4.3.1 Basic Decentralized EV Power Allocation

The objective function of the basic decentralized EV power allocation is shown in the following optimization problem:

$$\min_{\{V, L, r\}} \sum_{k=1}^K \sum_{g=1}^G [(A_{g,k} + V_{g,k}) - L_g]^2 \quad (4.8a)$$

$$\text{subject to } \sum_{k=1}^K V_{g,k} = \sum_{e \in \chi_g} P_c T_{min_e} - \sum_{h=1}^G r_{g,h} + \sum_{h=1}^G r_{h,g}, \forall g \in [1, G], \quad (4.8b)$$

$$\sum_{h=1}^G r_{g,h} \leq \sum_{e \in \chi_g} P_c T_{min_e}, \forall g \in [1, G], \quad (4.8c)$$

$$\sum_{h=1}^G r_{h,g} \leq B I_g, \forall g \in [1, G], \quad (4.8d)$$

$$\begin{cases} r_{g,h} = 0, & \text{when } g = h, \\ r_{g,h} \geq 0, & \text{when } g \neq h, \end{cases} \quad (4.8e)$$

$$0 \leq V_{g,k} \leq P G_{g,k}, \forall g \in [1, G], \forall k \in [1, K], \quad (4.8f)$$

$$L_g \geq 0, \forall g \in [1, G], \quad (4.8g)$$

where k and g represent the indexes of time slots and charging stations, respectively. The variable $A_{g,k}$ denotes the base load power, which is the power of all but EV loads, in sub-network g and time slot k . The EV load power for sub-network g at time slot k is denoted as $V_{g,k}$. The reference level for the sub-network g is a constant value L_g for time slots 1 to K . The term $r_{g,h}$ represents the EV energy flows from the charging station g to the charging station h during the period $[1, K]$ and is determined by the local station g . The set χ_g is the set of EVs requesting to be charged in sub-network g in the time window $[1, K]$. $\sum_{e \in \chi_g} P_c T_{min_e}$ is the overall energy for EV charging requests in charging station g . The upper bound of allocated EV power is denoted as \mathbf{PG} , where $P G_{g,k} = \min\{N m_g P_c, P t_g\}$. The

term, Nm_g , is the maximum number of EVs charging at station g , and Pt_g is the maximum EV power that the sub-network g can offer. The power flowing into any station should be less than an upper bound BI_g , the maximum EV energy from other stations that station g can accommodate during the period $[1, K]$. To make sure the problem has a solution, the following conditions should be met:

$$\sum_{g=1}^G \sum_{k=1}^K PG_{g,k} \geq \sum_{g=1}^G \sum_{e \in \chi_g} P_c T_{min_e}, \quad (4.9a)$$

$$\sum_{g=1}^G BI_g \geq 0. \quad (4.9b)$$

For the offline solution, load balancing for a period with length K (*i.e.*, $k \in [1, K]$) is studied. The charging station g will determine the variables of the allocated EV power $V_{g,k}$, the EV flowing power $r_{g,h}$ for all $h \in [1, G]$, and the reference level L_g by solving the optimization problem. Similar to the centralized objective function in Eq. (2.8), the deviation between the overall load and the reference level is expected to be minimized. However, different from the centralized problem, the reference levels can be different for each sub-network, and thus the deviation between the overall power and the reference level is minimized not only for time slots but also for sub-networks in this scenario. EV power allocated to the charging station, $V_{g,k}$, depends on three factors: overall requested EV charging power at the station, EV power shifted out from the station, and EV power shifted into the station. To simplify the model for the EV fleet, the constraints (4.8b) to (4.8d) regarding the EV

fleeting matrix \mathbf{r} are rewritten as the following constraints:

$$\sum_{k=1}^K V_{g,k} = \sum_{e \in \chi_g} P_c T_{min_e} - PO_g + PI_g, \forall g \in [1, G], \quad (4.10a)$$

$$PO_g = \sum_{h=1}^G r_{g,h}, \forall g \in [1, G], \quad (4.10b)$$

$$PI_g = \sum_{h=1}^G r_{h,g}, \forall g \in [1, G], \quad (4.10c)$$

$$0 \leq PO_g \leq \sum_{e \in \chi_g} P_c T_{min_e}, \forall g \in [1, G], \quad (4.10d)$$

$$0 \leq PI_g \leq BI_g, \forall g \in [1, G], \quad (4.10e)$$

where PO_g is the EV energy shifted out from station g in the period $[1, K]$, and PI_g is the EV energy shifted into the station g in the period $[1, K]$. The energy flow, PI_g and PO_g , is realized by EV fleets across sub-networks and bounded as shown in (4.10d) and (4.10e). Charging station g will optimize the EV energy that it shifts out to other stations for the purpose of balancing the power loads in the whole system. The EV energy flowing out the station, PO_g , corresponds to the EVs requested charging at station g but transferred to other stations and thus should be less than the energy sufficient for accommodating all charging requests at station g , *i.e.*, $\sum_{e \in \chi_g} P_c T_{min_e}$. The relationship between the energy flow from EV fleeting and overall power allocated at station g is shown in (4.10a). The overall energy flowing out from all G stations should be equal to the energy flowing into all stations, where $\sum_{g=1}^G PO_g = \sum_{g=1}^G PI_g$. After summing up the constraint (4.8b) for all g , the constraint (4.8b) or equivalently, (4.10a), the following equation can be obtained:

$$\sum_{k=1}^K \sum_{g=1}^G V_{g,k} = \sum_{g=1}^G \sum_{e \in \chi_g} P_c T_{min_e}. \quad (4.11)$$

The above equation requires that the overall power allocated for EV charging in the system should be equal to the overall charging power requested by EVs at all stations.

The objective function is convex. However, due to some constraints, such as (4.8b), the equation cannot be directly divided into subproblems for each charging station to solve locally. Proximal Jacobian ADMM is applied here to solve this problem. Since Proximal Jacobian ADMM methodology doesn't allow inequality constraints, inequality constraints (4.8e)-(4.8g), (4.10d) and (4.10e), are transformed into equivalent forms using auxiliary variable \mathbf{z}_i , where $i = 4, \dots, 8$, and \mathbf{z}_i is used in the i th constraint. Denote the vector/matrix of dual variables associated with the above eight constraints as \mathbf{U}_1 to \mathbf{U}_8 , respectively, which are defined as follows:

$$\mathbf{U}_c = \begin{cases} [U_{c,1}, U_{c,2}, \dots, U_{c,G}], & \text{for } c \neq 6 \text{ and } c \neq 7, \\ \begin{bmatrix} U_{c,1,1} & U_{c,1,2} & \dots & U_{c,1,G} \\ U_{c,2,1} & U_{c,2,2} & \dots & U_{c,2,G} \\ \vdots & \vdots & \ddots & \vdots \\ U_{c,G,1} & U_{c,G,2} & \dots & U_{c,G,G} \end{bmatrix}, & \text{for } c = 6, \\ \begin{bmatrix} U_{c,1,1} & U_{c,1,2} & \dots & U_{c,1,K} \\ U_{c,2,1} & U_{c,2,2} & \dots & U_{c,2,K} \\ \vdots & \vdots & \ddots & \vdots \\ U_{c,G,1} & U_{c,G,2} & \dots & U_{c,G,K} \end{bmatrix}, & \text{for } c = 7, \end{cases} \quad (4.12)$$

where the index $c = 1, \dots, 8$. The augmented Lagrangian $\Gamma(\mathbf{V}, \mathbf{L}, \mathbf{r}, \mathbf{PO}, \mathbf{PI}, \mathbf{z}, \mathbf{U})$ is sum-

marized in the equation below:

$$\begin{aligned}
\Gamma(\mathbf{V}, \mathbf{L}, \mathbf{r}, \mathbf{PO}, \mathbf{PI}, \mathbf{z}, \mathbf{U}) = & \sum_{k=1}^K \sum_{g=1}^G [(A_{g,k} + V_{g,k}) - L_g]^2 \\
& + \frac{\rho}{2} \sum_{g=1}^G (\bar{V}_g - \frac{1}{K} \sum_{e \in \chi_g} P_c T_{min_e} - \frac{1}{K} PO_g + \frac{1}{K} PI_g + U_{1,g})^2 + \frac{\rho}{2} \sum_{g=1}^G (\frac{1}{G} PO_g - \bar{r}_o + U_{2,g})^2 \\
& + \frac{\rho}{2} \sum_{g=1}^G (\frac{1}{G} PI_g - \bar{r}_{in} + U_{3,g})^2 + \frac{\rho}{2} \sum_{g=1}^G (PO_g - z_{4,g} + U_{4,g})^2 + \frac{\rho}{2} \sum_{g=1}^G (PI_g - z_{5,g} + U_{5,g})^2 \\
& + \frac{\rho}{2} \sum_{g=1}^G \sum_{h=1}^G (r_{g,h} - z_{6,g,h} + U_{6,g,h})^2 + \frac{\rho}{2} \sum_{g=1}^G \sum_{k=1}^K (V_{g,k} - z_{7,g,k} + U_{7,g,k})^2 \\
& + \frac{\rho}{2} \sum_{g=1}^G (L_g - z_{8,g} + U_{8,g})^2, \tag{4.13}
\end{aligned}$$

where $\bar{V}_g = \frac{1}{K} \sum_{k=1}^K V_{g,k}$, $\bar{r}_o = \frac{1}{G} \sum_{h=1}^G r_{g,h}$, $\bar{r}_{in} = \frac{1}{G} \sum_{h=1}^G r_{h,g}$. The terms, ρ and τ , are the parameters of Proximal Jacobian ADMM methodology.

The basic decentralized EV power allocation optimization by Proximal Jacobian ADMM algorithm is shown in the Algorithm 7. The function $\Pi_C(x)$ projects x into the subspace C . After variable splitting, only \bar{r}_{in} and \mathbf{PI} are the information not locally available to each sub-network for solving its subproblems. Thus, the coordinator collects information from each sub-network, calculates \bar{r}_{in} and \mathbf{PI} , and shares these variables with each charging station. In Algorithm 7, the EV power allocation $V_{g,k}$, the EV fleeing matrix $r_{g,h}$, and the reference level L_g can be determined locally by the charging station g without directly requesting load information from other sub-networks. The computation complexity for Algorithm 7 is analyzed as follows: in each iteration, all subproblems excepting (a) and (e) in Line 2 are solved K times. For the subproblem (a), $V_{g,k}$ is solved by each g and k in one iteration, and thus will be executed for GK times. For the subproblem (e), it will be computed for G^2 since the EV fleeing result is related to charging stations g and h .

Fig. 4.3 illustrates the algorithm processes for Algorithm 7. The example shown in Fig.

Algorithm 7 Proximal Jacobian ADMM Algorithm for Solving (4.8a)-(4.10e)

- 1: Iteration $i = 1$. Initial the variables $(\mathbf{V}, \mathbf{L}, \mathbf{r}, \mathbf{PO}, \mathbf{PI}, \mathbf{z}, \mathbf{U})^0 = (\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0})$, and $\Gamma(\mathbf{V}, \mathbf{L}, \mathbf{r}, \mathbf{PO}, \mathbf{PI}, \mathbf{z}, \mathbf{U})^0 = 0$ at epoch k .
 - 2: **for** Each bus $g = 1 : N$ **do** ▷ The step is done by charging stations in parallel.
 - (a) $V_{g,k}^i = \operatorname{argmin}_{V_{g,k}} [(V_{g,k} + A_{g,k}) - L_g^{i-1}]^2 + \frac{\rho}{2} (\bar{V}_g^{i-1} - V_{g,k}^{i-1} + V_{g,k} - \frac{1}{K} \sum_{e \in \chi_g} P_c T_{min_e} - \frac{1}{K} PO_g^{i-1} + \frac{1}{K} PI_g^{i-1} + U_{1,g}^{i-1})^2 + \frac{\rho}{2} (V_{g,k} - z_{7,g,k}^{i-1} + U_{7,g,k}^{i-1})^2 + \frac{\tau}{2} (V_{g,k} - V_{g,k}^{i-1})^2$, for $\forall k$.
 - (b) $\bar{V}_g^i = \frac{1}{K} \sum_{k=1}^K V_{g,k}^i$, $U_{3,h}^i = U_{3,h}^{i-1} + (\frac{1}{G} PI_h^{i-1} - \bar{r}_{in}^{i-1})$ for $\forall h \in [1, G]$.
 - (c) $PO_g^i = \operatorname{argmin}_{PO_g} \frac{\rho}{2} (\bar{V}_g^i - \frac{1}{K} \sum_{e \in \chi_g} P_c T_{min_e} - \frac{1}{K} PO_g + \frac{1}{K} PI_g^{i-1} + U_{1,g}^{i-1})^2 + \frac{\rho}{2} (\frac{1}{G} PO_g - \bar{r}_o^{i-1} + U_{2,g}^{i-1})^2 + \frac{\rho}{2} (PO_g - z_{4,g}^{i-1} + U_{4,g}^{i-1})^2 + \frac{\tau}{2} (PO_g - PO_g^{i-1})^2$.
 - (d) $PI_g^i = \operatorname{argmin}_{PI_g} \frac{\rho}{2} (\bar{V}_g^i - \frac{1}{K} \sum_{e \in \chi_g} P_c T_{min_e} - \frac{1}{K} PO_g^i + \frac{1}{K} PI_g + U_{1,g}^{i-1})^2 + \frac{\rho}{2} (\frac{1}{G} PI_g - \bar{r}_{in}^{i-1} + U_{3,g}^{i-1})^2 + \frac{\rho}{2} (PI_g - z_{5,g}^{i-1} + U_{5,g}^{i-1})^2 + \frac{\tau}{2} (PI_g - PI_g^{i-1})^2$.
 - (e) $r_{g,h}^i = \operatorname{argmin}_{r_{g,h}} \frac{\rho}{2} (\frac{1}{G} PO_g^i - \bar{r}_o^{i-1} - r_{g,h}^{i-1} + r_{g,h} + U_{2,g}^{i-1})^2 + \frac{\rho}{2} (\frac{1}{G} PI_h^{i-1} - \bar{r}_{in}^{i-1} - r_{g,h}^{i-1} + r_{g,h} + U_{3,h}^{i-1})^2 + \frac{\rho}{2} (r_{g,h} - z_{6,g,h}^{i-1} + U_{6,g,h}^{i-1})^2 + \frac{\tau}{2} (r_{g,h} - r_{g,h}^{i-1})^2$, for $\forall h = 1 \dots G$.
 - (f) $L_g^i = \operatorname{argmin}_{L_g} \sum_{k=1}^K [(V_{g,k} + A_{g,k}) - L_g]^2 + \frac{\rho}{2} (L_g - z_{8,g}^{i-1} + U_{8,g}^{i-1})^2 + \frac{\tau}{2} (L_g - L_g^{i-1})^2$.
 - (g) $z_{4,g}^i = \Pi_{C4}(PO_g^i + U_{4,g}^{i-1})$, $U_{4,g}^i = U_{4,g}^{i-1} + (PO_g^i - z_{4,g}^i)$, where $C4 = [0, \sum_{e \in \chi_g} P_c T_{min_e}]$.
 - (h) $z_{5,g}^i = \Pi_{C5}(PI_g^i + U_{5,g}^{i-1})$, $U_{5,g}^i = U_{5,g}^{i-1} + (PI_g^i - z_{5,g}^i)$, where $C5 = [0, BI_g]$.
 - (i) $z_{6,g,h}^i = \Pi_{C6}(r_{g,h}^i + U_{6,g,h}^{i-1})$, $U_{6,g,h}^i = U_{6,g,h}^{i-1} + (r_{g,h}^i - z_{6,g,h}^i)$, where $C6 = \mathbf{R}^+$ when $h \neq g$.
 - (j) $z_{7,g,k}^i = \Pi_{C7}(V_{g,k}^i + U_{7,g,k}^{i-1})$, $U_{7,g,k}^i = U_{7,g,k}^{i-1} + (V_{g,k}^i - z_{7,g,k}^i)$, where $C7 = [0, PG_{g,k}]$ for $\forall k$.
 - (k) $z_{8,g}^i = \Pi_{C8}(L_g^i + U_{8,g}^{i-1})$, $U_{8,g}^i = U_{8,g}^{i-1} + (L_g^i - z_{8,g}^i)$, where $C8 = \mathbf{R}$.
 - (l) $U_{1,g}^i = U_{1,g}^{i-1} + (\bar{V}_g^i - \frac{1}{K} \sum_{e \in \chi_g} P_c T_{min_e} - \frac{1}{K} PO_g^i + \frac{1}{K} PI_g^i)$.
 - (m) $r_o^i = \frac{1}{G} \sum_h r_{g,h}^i$, $U_{2,g}^i = U_{2,g}^{i-1} + (\frac{1}{G} PO_g^i - \bar{r}_o^i)$.
 - 3: **end for**
 - 4: The coordinator gathers the overall power flowing into charging stations PI_g^i and EV fleeing power among stations $r_{g,h}^i$. Then the coordinator distributes PI_g^i and the average of the power flowing into station g from other stations, $\bar{r}_{in}^i = \frac{1}{G} \sum_h r_{h,g}^i$.
 - 5: Find the result of Γ^i by Eq. (4.13).
 - 6: Repeat (2)-(5) until $|\Gamma^i - \Gamma^{i-1}| \leq \xi$.
-

4.3 is the power allocation for a station in the i th iteration. The EV energy from charging requests at Station 2 is simply denoted as E_2 in Fig. 4.3, where $E_2 = N_2 P_c T_{min}$. Charging stations upload the variable PI_g and $r_{g,h}$ to the coordinator (shown by the green dashed lines with arrows) and obtain \mathbf{PI} for others stations and \bar{r}_{in} from the coordinator (shown by the red solid lines with arrows) in each iteration. The local power allocation $V_{g,k}$, the base load information $A_{g,k}$, and the reference value L_g are local information to each sub-networks. In

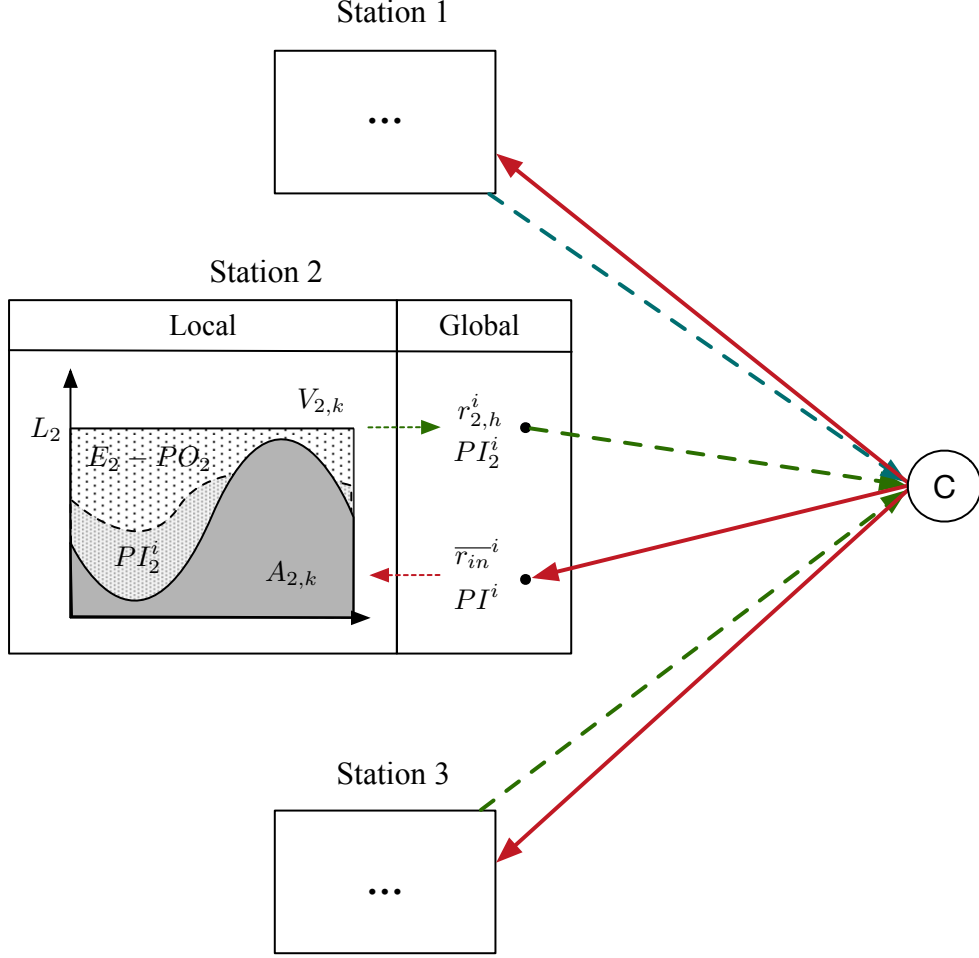


Figure 4.3: Illustration for Algorithm 7, where $E_2 = \sum_{e \in \chi_2} P_c T_{min_e}$.

each iteration, load balancing is achieved locally considering the energy flowing among the stations and the energy of EV charging requests at the station g . The valley of base load $A_{g,k}$ is filled by the EV energy from the accepted EV requests $\sum_{e \in \chi_g} P_c T_{min_e} - PO_g$ and the energy shifted to station g from other stations, *i.e.*, PI_g .

Load balancing among sub-networks and inside sub-networks is achieved simultaneously by ADMM optimization. ADMM algorithm is directly fitted into the proposed problem, which inherits the objective of the centralized approach for the proposed problem. In our proposed optimization problem, the EV power allocation is constrained by overall EV energy

in the system, as shown in (4.11) regardless of how EV energy flows inside the system. The value of PI_g represents EV energy from other sub-networks to balance the loads. Besides, all sub-networks agree on the same EV fleeting matrix \mathbf{r} at the output of their local optimization process.

The basic EV power allocation can achieve the load balancing for every duration with K time slots in a decentralized manner under the assumption that EVs are fully elastic with the same power budget. However, the EV arrivals and charging time requirements may vary depending on the EV status. Therefore, a real-time EV power allocation scheme is introduced in the next section for achieving the load balancing with dynamic EV arrivals.

4.3.2 Dynamic Decentralized EV Power Allocation

In the dynamic EV power allocation scheme, we split EV power allocation at station g and time slot k into two parts: the power for charging EVs arrived in slot k at station, denoted as $Q_{g,k}$, and the power for charging EVs arrived in previous time slots and yet to complete charging, denoted as $V_{g,k}$. Furthermore, the power for charging EVs arrived in previous time slots, $V_{g,k}$, concerns two cases: vehicles that have started charging may or may not be disrupted depending on its Tc_e . Once EV e is scheduled to be charged, it will be charged continuously by t_c time slots. The term $Vp_{g,k}$ represents the power necessary to be allocated to vehicles that have already started charging and need to be charged without disruption in order to complete their charging process in time. Therefore, $V_{g,k}$ should be no less than $Vp_{g,k}$. A real-time EV allocation for the dynamic EV fleeting is proposed in this subsection.

The new objective function can be written as:

$$\min_{\{\mathbf{V}, \mathbf{Q}, \mathbf{L}, \mathbf{r}\}} \sum_{g=1}^G [(V_{g,k} + A_{g,k} + Q_{g,k}) - L_{g,k}]^2 + \mu \sum_{g=1}^G (L_{g,k} - Lpre_{g,k})^2 \quad (4.14a)$$

$$\text{subject to } Q_{g,k} = \sum_{e \in \chi_{g,k}} P_c + \sum_{h=1}^G r_{h,g,k} - \sum_{h=1}^G r_{g,h,k}, \quad (4.14b)$$

$$\sum_{h=1}^G r_{g,h,k} \leq \sum_{e \in \chi_{g,k}} P_c, \quad (4.14c)$$

$$\sum_{h=1}^G r_{h,g,k} \leq BI_{g,k}, \quad (4.14d)$$

$$\begin{cases} r_{g,h,k} = 0, & \text{when } g = h, \\ r_{g,h,k} \geq 0, & \text{when } g \neq h, \end{cases} \quad (4.14e)$$

$$0 \leq Q_{g,k} + V_{g,k} \leq PG_{g,k}, \quad (4.14f)$$

$$Vp_{g,k} \leq V_{g,k} \leq Nc_{g,k}P_c, \quad (4.14g)$$

$$Q_{g,k} \geq 0, L_{g,k} \geq 0, \quad (4.14h)$$

where $\chi_{g,k}$ is the set of new EVs charging requests at station g and time slot k . The term $L_{g,k}$ denotes the reference level for the sub-network g in time slot k . $Lpre_{g,k}$ is obtained from the power profile information in previous stages. In this work, $Lpre_{g,k} = \frac{1}{T_w} \sum_{l=k-T_w}^{k-1} (V_{g,l} + Q_{g,l} + A_{g,l})$, where T_w is the time window length. The term, $r_{g,h,k}$, represents the EV power flowing out from station g to h in the k th time slot. The term $Nc_{g,k}P_c$ is the maximum power when all EVs plugged into the station are fully charged in the current time slot, where $Nc_{g,k}$ is the number of the EVs that arrived at charging station g no later than time slot $k - 1$.

Similar to the online centralized load balancing problem, the reference level \mathbf{L}_g for a sub-network can be time-index dependent rather than a constant value to accommodate the different EV power budget and the base load power in real-time. Meanwhile, EV power

fleeting matrix, \mathbf{r} , varies in different time slots due to the dynamic circulation of EVs as well. In the main objective function, the first part minimizes the deviation of overall power demand from the reference level in a time slot in the sub-network. Similar to the constraint (4.8f), the overall EV power in a station in the k th slot, $Q_{g,k}$ plus $V_{g,k}$, should not exceed the elastic load upper bound $PG_{g,k}$. The power of newly accepted EVs $Q_{g,k}$ is related with the power from EV charging requests in the sub-network and the power flow among the stations in the current time slot. The power allocation of EVs already plugged in the station, $V_{g,k}$, is bounded between $Vp_{g,k}$ and $Nc_{g,k}P_c$. Similar to the offline case, the constraints regarding EV fleeting are rewritten as EV power flow \mathbf{PI} and \mathbf{PO} , which are shown in following equations:

$$Q_{g,k} = \sum_{e \in \chi_{g,k}} P_c + PI_{g,k} - PO_{g,k}, \quad (4.15a)$$

$$PO_{g,k} = \sum_{h=1}^G r_{g,h,k}, \quad (4.15b)$$

$$PI_{g,k} = \sum_{h=1}^G r_{h,g,k}, \quad (4.15c)$$

$$0 \leq PO_{g,k} \leq \sum_{e \in \chi_{g,k}} P_c, \quad (4.15d)$$

$$0 \leq PI_{g,k} \leq BI_{g,k}, \quad (4.15e)$$

where $PO_{g,k}$ and $PI_{g,k}$ represent the overall power flowing out and into the station g in the k th time slot, respectively.

The second part of the objective function minimizes the fluctuation of the reference level compared with the power demand history. The reference level in the current time slot, $L_{g,k}$, is compared with the parameter $Lpre_{g,k}$. The factor μ controls the relative significance of the power fluctuation to the power grid while solving the considered optimization problem.

The power demand is expected to be smoother with a higher μ value. Compared with the objective function (4.8a), the objective function for the dynamic problem is optimized in each time slot such that load balancing can be achieved given current EV information.

The objective function (4.14a) is transformed to an augmented Lagrangian,

$$\Theta(\mathbf{V}, \mathbf{L}, \mathbf{r}, \mathbf{PO}, \mathbf{PI}, \mathbf{Q}, \mathbf{z}, \mathbf{U}),$$

where the scaled dual variables \mathbf{U}_1 to \mathbf{U}_{10} are defined as follows:

$$\mathbf{U}_c = \begin{cases} [U_{c,1}, U_{c,2}, \dots, U_{c,G}], & \text{for } c \neq 6, \\ \begin{bmatrix} U_{c,1,1} & U_{c,1,2} & \dots & U_{c,1,G} \\ U_{c,2,1} & U_{c,2,2} & \dots & U_{c,2,G} \\ \vdots & \vdots & \ddots & \vdots \\ U_{c,G,1} & U_{c,G,2} & \dots & U_{c,G,G} \end{bmatrix}, & \text{for } c = 6, \end{cases} \quad (4.16)$$

where $c = 1, \dots, 10$. The augmented Lagrangian is shown as follows:

$$\begin{aligned} \Theta(\mathbf{V}, \mathbf{L}, \mathbf{r}, \mathbf{PO}, \mathbf{PI}, \mathbf{Q}, \mathbf{z}, \mathbf{U}) = & \sum_{g=1}^G [(V_{g,k} + A_{g,k} + Q_{g,k}) - L_{g,k}]^2 + \mu \sum_{g=1}^G (L_{g,k} - L_{pre,g,k})^2 \\ & + \frac{\rho}{2} \sum_g \left(\sum_{e \in \chi_{g,k}} P_c + PI_{g,k} - PO_{g,k} - Q_{g,k} + U_{1,g} \right)^2 + \frac{\rho}{2} \sum_g \left(\frac{PO_{g,k}}{G} - \bar{r}_{out} + U_{2,g} \right)^2 \\ & + \frac{\rho}{2} \sum_g \left(\frac{PI_{g,k}}{G} - \bar{r}_{in} + U_{3,g} \right)^2 + \frac{\rho}{2} \sum_g (PO_{g,k} - z_{4,g} + U_{4,g})^2 \\ & + \frac{\rho}{2} \sum_g (PI_{g,k} - z_{5,g} + U_{5,g})^2 + \frac{\rho}{2} \sum_{g,h} (r_{g,h,k} - z_{6,g,h} + U_{6,g,h})^2 \\ & + \frac{\rho}{2} \sum_g (V_{g,k} + Q_{g,k} - z_{7,g} + U_{7,g})^2 + \frac{\rho}{2} \sum_g (V_{g,k} - z_{8,g} + U_{8,g})^2 \\ & + \frac{\rho}{2} \sum_g (Q_{g,k} - z_{9,g} + U_{9,g})^2 + \frac{\rho}{2} \sum_g (L_{g,k} - z_{10,g} + U_{10,g})^2 \end{aligned} \quad (4.17)$$

Then the dynamic EV allocation problem is split to subproblems by Proximal Jacobian

ADMM technique, summarized in Algorithm 8.

Algorithm 8 Proximal Jacobian ADMM Algorithm for Solving (4.14a)-(4.15e)

- 1: Iteration $i = 1$. Initial the variables $\Theta(\mathbf{V}, \mathbf{L}, \mathbf{r}, \mathbf{PO}, \mathbf{PI}, \mathbf{Q}, \mathbf{z}, \mathbf{U})^0 = \mathbf{0}$, and $(\mathbf{V}, \mathbf{L}, \mathbf{r}, \mathbf{PO}, \mathbf{PI}, \mathbf{Q}, \mathbf{z}, \mathbf{U})^0 = (\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0})$ at epoch k .
 - 2: **for** Each bus $g = 1 : G$ **do** ▷ The step is done by charging stations in parallel.
 - (a) $V_{g,k}^i = \operatorname{argmin}_{V_{g,k}} [(V_{g,k} + A_{g,k} + Q_{g,k}^{i-1}) - L_{g,k}^{i-1}]^2 + \frac{\rho}{2} [(V_{g,k} + Q_{g,k}^{i-1} - z_{g,k}^{i-1}) + U_{7,g}^{i-1}]^2 + \frac{\rho}{2} [(V_{g,k} - z_{8,g}^{i-1}) + U_{8,g}^{i-1}]^2 + \frac{\tau}{2} (V_{g,k} - V_{g,k}^{i-1})^2$.
 - (b) $Q_{g,k}^i = \operatorname{argmin}_{Q_{g,k}} [(V_{g,k}^i + A_{g,k} + Q_{g,k}) - L_{g,k}^{i-1}]^2 + \frac{\rho}{2} [(\sum_{e \in \chi_{g,k}} P_c + PI_{g,k}^{i-1} - PO_{g,k}^{i-1} - Q_{g,k}) + U_{1,g}^{i-1}]^2 + \frac{\rho}{2} [(V_{g,k}^i + Q_{g,k} - z_{7,g}^{i-1}) + U_{7,g}^{i-1}]^2 + \frac{\rho}{2} [(Q_{g,k} - z_{9,g}^{i-1}) + U_{9,g}^{i-1}]^2 + \frac{\tau}{2} (Q_{g,k} - Q_{g,k}^{i-1})^2$.
 - (c) $U_{3,h}^i = U_{3,h}^{i-1} + (\frac{1}{G} PI_{h,k}^{i-1} - \bar{r}_{in}^{i-1})$, $\forall h \in [1, G]$
 - (d) $PO_{g,k}^i = \operatorname{argmin}_{PO_{g,k}} \frac{\rho}{2} [(\sum_{e \in \chi_{g,k}} P_c + PI_{g,k}^{i-1} - PO_{g,k} - Q_{g,k}^i) + U_{1,g}^{i-1}]^2 + \frac{\rho}{2} [(\frac{1}{G} PO_{g,k} - \bar{r}_{out}^{i-1}) + U_{2,g}^{i-1}]^2 + \frac{\rho}{2} [(PO_{g,k} - z_{4,g}^{i-1}) + U_{4,g}^{i-1}]^2 + \frac{\tau}{2} (PO_{g,k} - PO_{g,k}^{i-1})^2$.
 - (e) $PI_{g,k}^i = \operatorname{argmin}_{PI_{g,k}} \frac{\rho}{2} [(\sum_{e \in \chi_{g,k}} P_c + PI_{g,k} - PO_{g,k}^{i-1} - Q_{g,k}^i) + U_{1,g}^{i-1}]^2 + \frac{\rho}{2} [(\frac{1}{G} PI_{g,k} - \bar{r}_{in}^{i-1}) + U_{3,g}^{i-1}]^2 + \frac{\rho}{2} [(PI_{g,k} - z_{5,g}^{i-1}) + U_{5,g}^{i-1}]^2 + \frac{\tau}{2} (PI_{g,k} - PI_{g,k}^{i-1})^2$.
 - (f) $r_{g,h,k}^i = \operatorname{argmin}_{r_{g,h,k}} \frac{\rho}{2} [(\frac{1}{G} PO_{g,k}^i - \bar{r}_{out}^{i-1} + r_{g,h,k}^{i-1} - r_{g,h,k}) + U_{2,g}^{i-1}]^2 + \frac{\rho}{2} [(\frac{1}{G} PI_{h,k}^{i-1} - \bar{r}_{in}^{i-1} + r_{g,h,k}^{i-1} - r_{g,h,k}) + U_{3,h}^{i-1}]^2 + \frac{\rho}{2} [(r_{g,h,k} - z_{6,g,h}^{i-1}) + U_{6,g,h}^{i-1}]^2 + \frac{\tau}{2} (r_{g,h,k} - r_{g,h,k}^{i-1})^2$, $\forall n = 1 \dots G$.
 - (g) $z_{6,g,h}^i = \Pi_{C6}(r_{g,h,k}^i + U_{66,g,h}^{i-1})$, $U_{6,g,h}^i = U_{6,g,h}^{i-1} + (r_{g,h,k}^i - z_{6,g,h}^i)$, where $C7 = \mathbf{R}^+$, if $h \neq g$.
 - (h) $L_{g,k}^i = \operatorname{argmin}_{L_{g,k}} [(V_{g,k}^i + A_{g,k} + Q_{g,k}^i) - L_{g,k}]^2 + \mu \sum_{g=1}^G (L_{g,k} - L_{pre,g,k})^2 + \frac{\rho}{2} \sum_g [(L_{g,k} - z_{10,g}^{i-1}) + U_{10,g}^{i-1}]^2 + \frac{\tau}{2} (L_{g,k} - L_{g,k}^{i-1})^2$.
 - (i) $U_{1,g}^i = U_{1,g}^{i-1} + (\sum_{e \in \chi_{g,k}} P_c + PI_{g,k}^i - PO_{g,k}^i - Q_{g,k}^i)$.
 - (j) $r_{out}^i = \frac{1}{G} \sum_h r_{g,h,k}^i$, $U_{2,g}^i = U_{2,g}^{i-1} + (\frac{1}{G} PO_{g,k}^i - \bar{r}_{out}^i)$.
 - (k) $z_{4,g}^i = \Pi_{C4}(PO_{g,k}^i + U_{4,g}^{i-1})$, $U_{4,g}^i = U_{4,g}^{i-1} + (PO_{g,k}^i - z_{4,g}^i)$, where $C4 = [0, \sum_{e \in \chi_{g,k}} P_c]$.
 - (l) $z_{5,g}^i = \Pi_{C5}(PI_{g,k}^i + U_{5,g}^{i-1})$, $U_{5,g}^i = U_{5,g}^{i-1} + (PI_{g,k}^i - z_{5,g}^i)$, where $C5 = [0, BI_{g,k}]$.
 - (m) $z_{7,g}^i = \Pi_{C7}(V_{g,k}^i + Q_{g,k}^i + U_{7,g}^{i-1})$, $U_{7,g}^i = U_{7,g}^{i-1} + (V_{g,k}^i + Q_{g,k}^i - z_{7,g}^i)$, where $C5 = [0, PG_g]$.
 - (n) $z_{8,g}^i = \Pi_{C8}(V_{g,k}^i + U_{8,g}^{i-1})$, $U_{8,g}^i = U_{8,g}^{i-1} + (V_{g,k}^i - z_{8,g}^i)$, where $C8 = [Vp_{g,k}, Nc_g * P_c]$.
 - (o) $z_{9,g}^i = \Pi_{C9}(Q_{g,k}^i + U_{9,g}^{i-1})$, $U_{9,g}^i = U_{9,g}^{i-1} + (Q_{g,k}^i - z_{9,g}^i)$, where $C8 = \mathbf{R}^+$.
 - (p) $z_{10,g}^i = \Pi_{C10}(L_{g,k}^i + U_{10,g}^{i-1})$, $U_{10,g}^i = U_{10,g}^{i-1} + (L_{g,k}^i - z_{10,g}^i)$, where $C10 = \mathbf{R}^+$.
 - 3: **end for**
 - 4: The coordinator gathers overall power flowing into the station $PI_{g,k}^i$ and the detailed EV fleeing power among stations $r_{h,g,k}^i$. Then the coordinator distributes the overall and the mean of EV power flowing into the station g from other stations: $PI_{g,k}^i$ and \bar{r}_{in}^i , where $\bar{r}_{in}^i = \frac{1}{G} \sum_h r_{h,g,k}^i$.
 - 5: Find the result of Θ^i by Eq. (4.17).
 - 6: Repeat (2)-(5) until $|\Theta^i - \Theta^{i-1}| \leq \xi$.
-

Through Proximal Jacobian ADMM technique, sub-networks allocate EV power given

the load information in real-time and adjust the reference level according to local power information. The coordinator only gathers the EV fleeting information, \mathbf{r} , and then calculates and broadcasts the mean value $\overline{r_{in}}$. The subproblem (f) in Line 2 will be solved by GK times for one iteration, while other subproblems are solved for G times in Algorithm 8.

Fig. 4.4 illustrates Algorithm 8 for a charging station in a time slot with dynamic EV arrivals. The EV power allocation for current time slot contains the power of EV charging requests accepted in this time slot, $Q_{g,k}$, and the power for the EVs arrived prior to the current time slot, $V_{g,k}$. In each iteration, \mathbf{PI} and \bar{r} are collected and distributed among charging stations to achieve the global optimization by local load balancing.

The proposed dynamic power allocation is developed based on the basic EV power allocation scheme. Similar to the case of basic decentralized allocation, the variable splitting reduces information exchanging and protects the local information. Unlike the basic model, the dynamic reference level adjustment is considered similar to the case in the centralized dynamic load balancing strategy. The dynamic decentralized EV power allocation scheme can satisfy the time elasticity of EVs in real-time since it monitors the EV status in each time slot.

In the next section, an EV random charging scheme is introduced for allocating the EV units to implement the EV power allocation result obtained in this section. EVs can be fully charged with constant charging power with a time limitation while achieving the load balancing objective.

4.4 EV Random Charging Scheme: Implementation of Optimal Power Allocation

The proposed algorithms in the previous section derive EV power allocation and energy sharing among sub-network to achieve load balancing. Due to limited EV power, not all EVs can start their charging process immediately after their arrival, and charging for a part

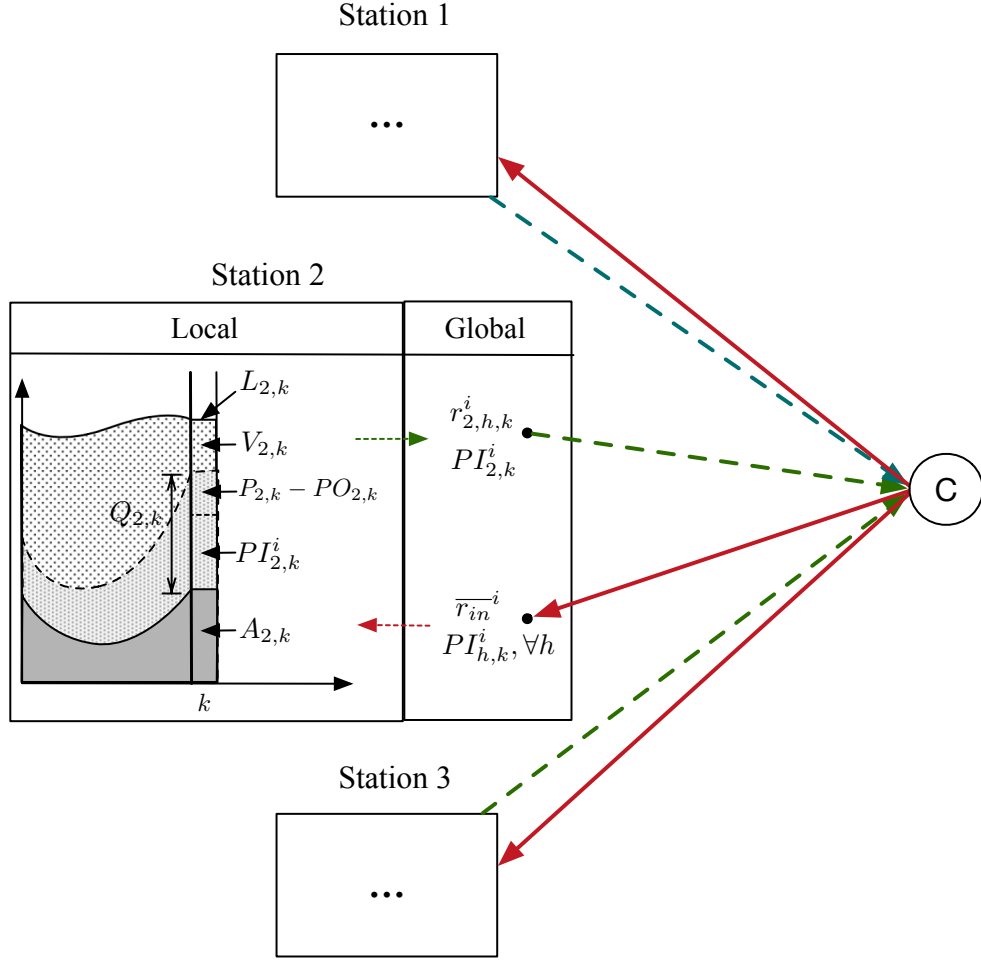


Figure 4.4: Illustration for Algorithm 8, where $P_{2,k} = \sum_{e \in \chi_{2,k}} P_c$.

of EVs may be interrupted. In this section, in order to achieve load balancing by EVs in the system, a random charging scheme is proposed to implement the optimal EV power obtained from the optimization problem and manage the EVs to follow the power target as well as satisfy charging time requirements for each individual EV. With randomness in the scheme, EVs plug into the system based on a pre-calculated probability so that the optimal power allocation derived by charging stations can be implemented while fairness among EVs can be achieved. Newly arrived EV units will be accepted or transferred to other charging stations according to EV fleeting results. After an EV plugged in, the proposed random

charging scheme manages its charging process with/without delay depending on its charging time requirement. Using the proposed scheme, EV units can be fairly scheduled to satisfy the optimal EV power allocation without violating the allocated EV power.

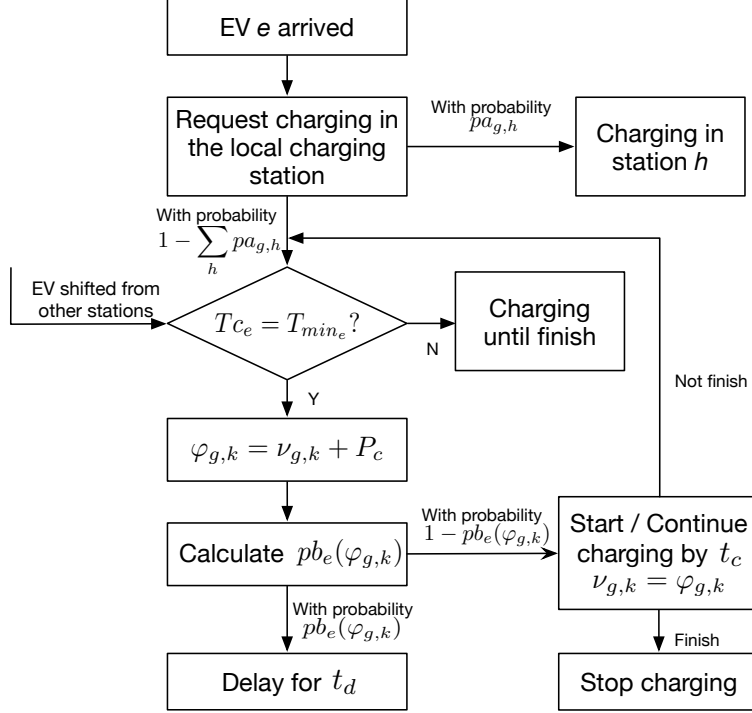


Figure 4.5: EV charging scheme flow chart.

Fig. 4.5 presents the scheduling procedure for the e th EV. At first, EV e enters the zone for the local charging station g and requests charging in time slot k . Then, the charging station g determines whether accepting or shifting the request based on the probability $pa_{g,h}$, which can be obtained from the EV power optimization results. The equation for $pa_{g,h}$ is shown as follows:

$$pa_{g,h} = \begin{cases} \frac{r_{g,h}}{\sum_{e \in \chi_g} P_c T_{min_e}}, & \text{for the basic scheme,} \\ \frac{r_{g,h,k}}{\sum_{e \in \chi_{g,k}} P_c}, & \text{for the dynamic scheme.} \end{cases} \quad (4.18)$$

where $g \neq h$. The probability represents the ratio of the power for transferring EV to

another station with the power for current EVs requests in the station. With the probability $pa_{g,h}$, EV e will be requested to transfer to the charging station h . Otherwise, the station g will accept the EV e 's request and evaluate the time flexibility of the EV by comparing its minimum charging time T_{min_e} and current charging time till the EV leaving T_{c_e} . For the basic offline scheme, T_{c_e} is evaluated as the time length from the current slot k to K . For the dynamic allocation scheme, T_{c_e} may vary with each specific EVs. If $T_{c_e} = T_{min_e}$, EV e has to be charged without being interrupted until its battery is full to satisfy the charging requirement. Correspondingly, the lower bound constraint $Vp_{g,t}$ should be reassigned with the value $Vp_{g,t} + P_c$ for $t \in [k, k + T_{min_e} - 1]$ for every case of a vehicle requiring continuous charging without interruption. If $T_{c_e} > T_{min_e}$, the station will continue further scheduling since EV e is able to be fully charged with at least one time slot of delay. The variable, $\nu_{g,k}$ in Fig. 4.5, is the EV power already been scheduled at station g in time slot k . The initial value for $\nu_{g,k}$ is defined as $Vp_{g,k}$. The variable $\varphi_{g,k}$ denotes the EV power if the station g charges EV e in time slot k . Afterwards, an exponential model proposed by [56] is modified in this work to determine the delay of charging with the blocking probability $pb_e(\varphi_{g,k})$. The $pb_e(\varphi_{g,k})$ is summarized in the following equation:

$$pb_e(\varphi_{g,k}) = \begin{cases} \min\{e^{\alpha(\varphi_{g,k} - W_{g,k}) - \beta(T_{min_e}/T_{c_e})}, 1\}, & \text{if } T_{c_e} > T_{min_e}, \\ 0, & \text{if } T_{c_e} \leq T_{min_e}, \end{cases} \quad (4.19)$$

where $W_{g,k} = V_{g,k}$ for the basic offline scheme and $W_{g,k} = V_{g,k} + Q_{g,k}$ for the dynamic EV schedule scheme. The parameter α , a trade-off factor between the deviation from the optimal result. The parameter β represents the impact of EV charging time flexibility on the blocking probability. EVs with less time flexibility tend to have a low probability of getting blocked when β is high. To find the factor α , it is assumed that $\beta = 0$ at first. When $\varphi_{g,k} \geq V_{g,k}$ for the basic scheme or $\varphi_{g,k} \geq V_{g,k} + Q_{g,k}$ for the dynamic case, the probability $pb_e(\varphi_{g,k}) = 1$ since no elastic power can be scheduled from the given EV power result. The parameter Φ is introduced to determine α . When $\varphi_{g,k} = \Phi$, the blocking probability $pb_e(\Phi)$ is ϵ . Then,

the factor α can be defined in the following equation:

$$\alpha = \frac{\ln \epsilon}{\Phi_{g,k} - W_{g,k}}. \quad (4.20)$$

Fig. 4.6 presents the evaluation for α . Finally, if the charging station g blocks the charging request by the blocking probability $pb_e(\varphi_{g,k})$, the EV will request to be charged again after t_d time slots, where $t_d < T_{c_e} - T_{min_e}$. Otherwise, $\nu_{g,k}$ is updated, and EV e will be charged consistently with the constant power P_c for t_c time slots, where $t_c \leq T_{min_e}$.

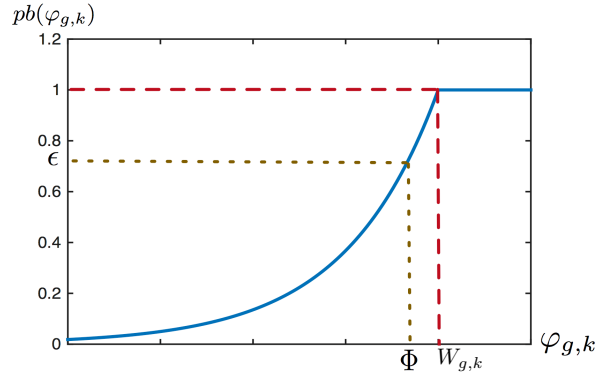


Figure 4.6: The blocking probability $pb(\varphi_{g,k})$ vs. $\varphi_{g,k}$.

4.5 Performance Evaluation

4.5.1 Simulation Results of Basic Decentralized EV Power Allocation Scheme

In this section, numerical results are given for the basic decentralized EV power allocation scheme in the offline scenario, where the load information for a day is known. The base load data is from [2]. The time slot length is one hour, *i.e.*, $K = 24$. There are five sub-networks (buses) considered in the simulation. For Buses 1 and 2, there are 100 residential units, Buses 3 and 4 with 25 retail units, and Bus 5 with 25 industrial units. For the basic

allocation scheme, EVs are assumed to enter buses in the first time slot with full elasticity on time. The number of EVs entering Buses 1 to 3 is $\{150, 150, 60\}$, and there is no EV entering the Buses 4 and 5. The charging rate P_c is 3 kW, and the minimum charging time T_{min_e} is 6 hours for all EVs. The upper bound BI_g is infinity. The parameter setting for the random charging scheme is $\Phi_{g,k} = 0.8V_{g,k}$ and $\epsilon = 0.1$, namely, when $\Phi_{g,k}$ achieves 80% of the allocated EV power, the blocking percentage is 10% .

Fig. 4.7 depicts the power demand scheduled by the proposed basic decentralized EV power allocation scheme in five buses and the overall system without considering the upper bound of elastic load power, $PG_{g,k}$. The EV number in Buses 1 to 5 after EV fleeing is $\{83, 73, 48, 89, 67\}$ respectively. Load balancing for Buses 4 and 5 are achieved by EVs fed by other buses since there is no EV load to fill the valley of the power demand by base load without EV fleeing. The standard deviations of the load over the 5 buses are $[2.51, 1.84, 2.81, 5.53, 3.36]$, respectively. A small fluctuation is caused by the high base load and the uncertainty on the random allocation scheme.

Furthermore, Fig. 4.8 shows the power demand profile for five buses and overall system with the appropriated ***PG*** setting, where ***PG*** is $\{\text{Inf}, \text{Inf}, 200, 100, 0\}$ for the five buses in a day. As shown in the Figs. 4.8(a) and (b), compared with Figs. 4.7(a) and (b), more EV loads are allocated in Buses 1 and 2 since EV power in other buses is constrained by finite ***PG***. The constraint of ***PG*** impacts Buses 4 and 5 significantly. The EV power is bounded by 100 kW for Bus 4, and no EV power is allocated on Bus 5.

Fig. 4.9 shows the overall power demand comparison for three cases: the proposed approach without EV fleets, with EV fleets and finite ***PG*** setting, and with EV fleets and infinite ***PG*** setting. The power fluctuation is the highest among three circumstances with the standard deviation 91.91 because load balancing is hard to be achieved with less EV power in some buses, especially for the Buses 4 and 5. Furthermore, the power demand with EV fleeing and infinite ***PG*** shows the smallest fluctuation among those cases, and the power demand with EV fleeing and finite ***PG*** shows more fluctuation as expected for

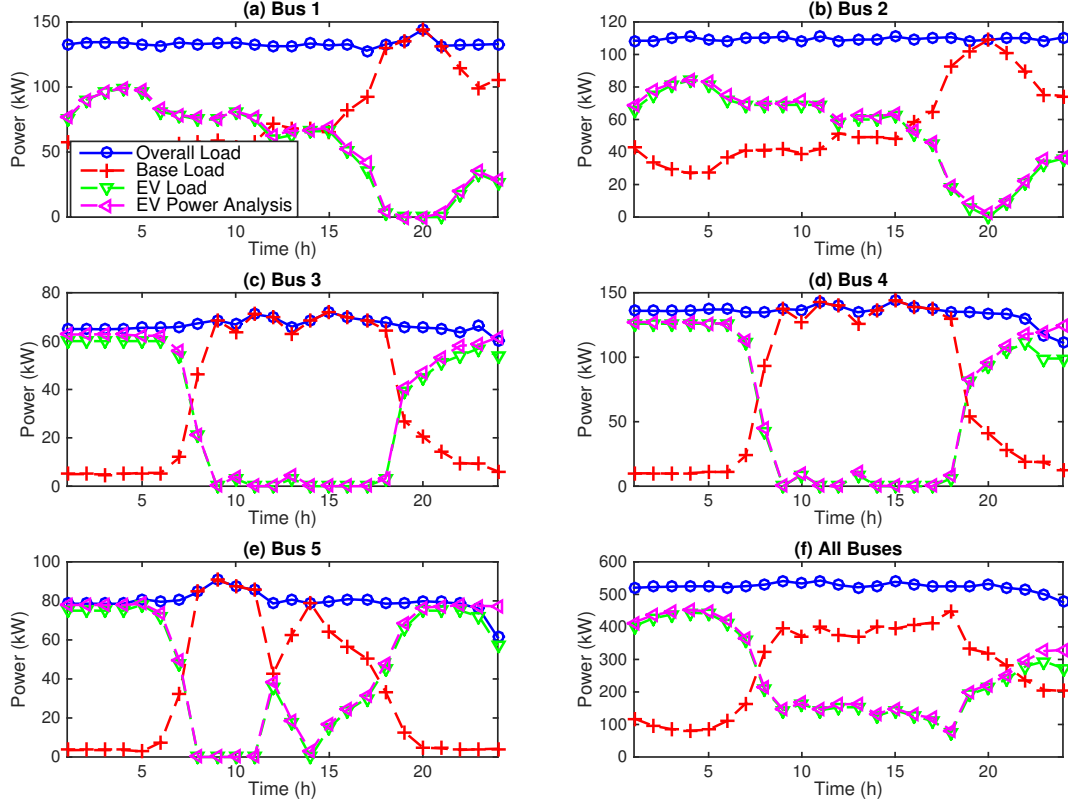


Figure 4.7: The power demand profile for EV loads and base loads among 5 buses and the whole system by the basic decentralized EV power allocation scheme in one day. PG is infinity.

the constrained EV power. Overall, by our method, the optimization process for the whole system is split into several sub-networks. Meanwhile, the optimization in sub-networks contributes to fluctuation reduction for the whole system.

4.5.2 Simulation Results of Dynamic Decentralized EV Power Allocation Scheme

On top of the basic EV load allocation scheme, the dynamic EV load allocation can schedule dynamic EV arrivals and satisfy EV users' requirements in the real-time. The parameter

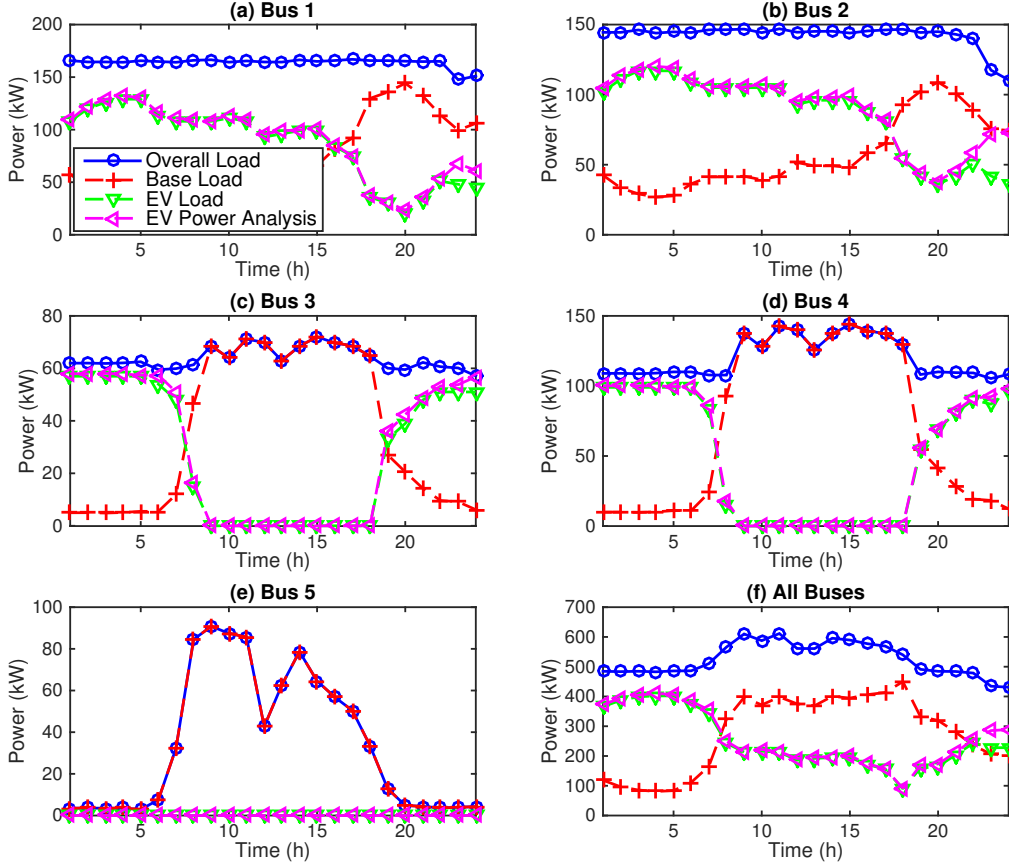


Figure 4.8: The power profile for EV loads and base loads among 5 buses and the whole system by Proximal Jacobian ADMM and random charging scheme in one day with finite ***PG***.

setting for base loads is the same as the simulation setting of the basic EV load allocation. The number of EV entering Buses 1 to 3 is $\{300, 300, 120\}$, and no EVs enter Buses 4 and 5. The future demand for EV load and the future base load are unknown when the algorithm runs in real-time. The more dynamic the system is, the more charging requirements are considered in the simulation: EVs entering the charging station is uniformly distributed in a day. The energy demand for each EV is uniformly distributed between $[1, 45]$ kWh. T_{c_e} is long enough for the EV to be fully charged, which is uniformly distributed between

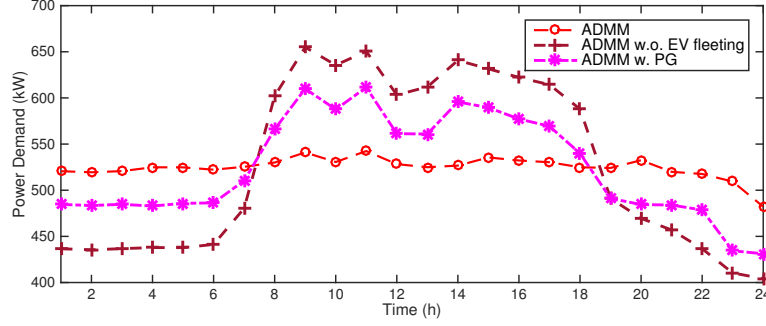


Figure 4.9: The comparison of overall power demand scheduled without EV fleetings, with EV fleetings and **PG** constraint, and without **PG** constraint.

$[Tmin_i+12, 48]$ hours. The initial reference value $Lpre_{g,k}$ is initiated with the power demand history of the unscheduled case. The reference level weight is $\mu = 2$.

Fig. 4.10 shows the comparison between the unscheduled demand and overall power demand scheduled by the proposed ADMM power allocation and random scheduling algorithm (ADMM+RS) in 6 days. Since the initial reference levels are the unscheduled demand for each sub-network, the reference levels are adjusted in 6 days to approach the optimal value. The weight μ can adjust the rate of the reference level adjustment. The reference level leads the power allocation to flatten the power demand. Compared with the unscheduled case, the fluctuation of power demand is reduced by load balancing in real-time.

The performance of power demand scheduled by the proposed algorithm in the 6th day is shown in Fig. 4.11. EVs fill the valley of the base load power and circulate out when the base load dominates. The standard deviation is reduced to [1.21; 0.74; 1.15; 14.97; 9.98] for five buses, while standard deviations for the unscheduled case are [38.39; 44.85; 26.83; 51.67; 34.59]. Correspondingly, the overall power fluctuation is reduced from 123.67 to 21.55 with the proposed approach.

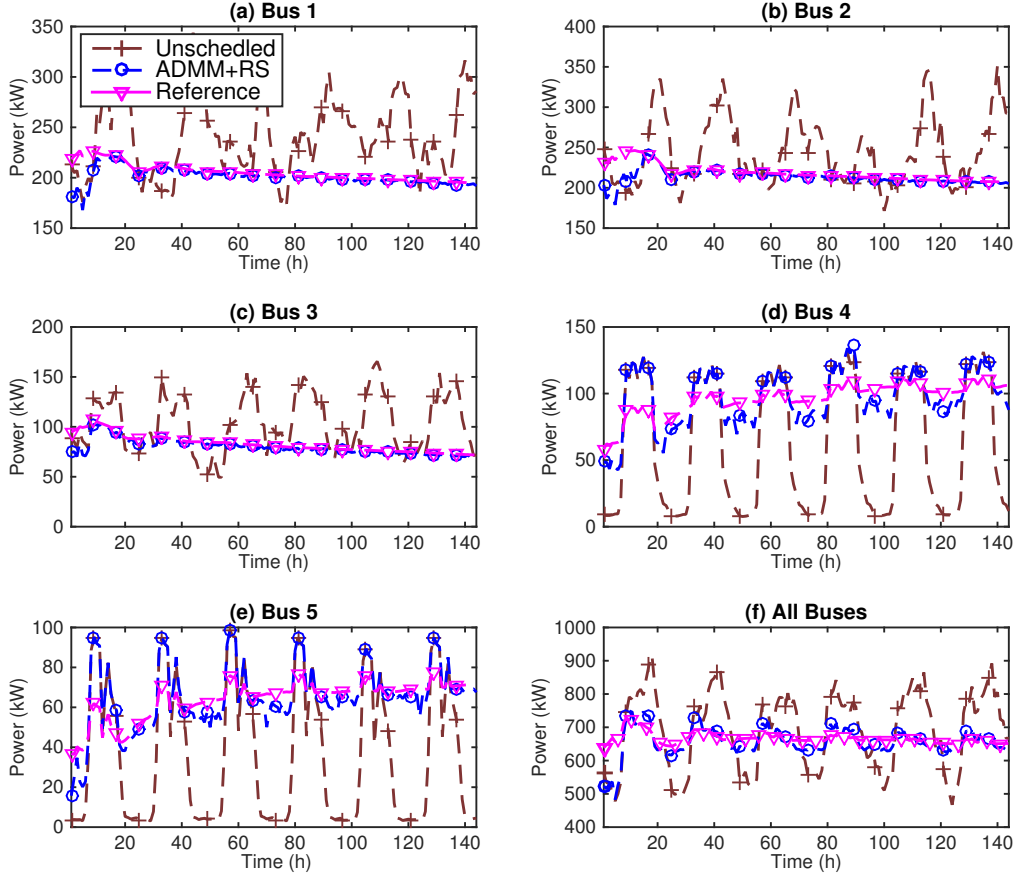


Figure 4.10: The overall power profile for 5 buses and the whole system for unscheduled and scheduled cases in 6 days.

4.6 Chapter Summary

In this chapter, decentralized load balancing strategy among the sub-networks is studied. The objective is inherited from centralized power allocation approaches in the previous chapters. Proximal Jacobian ADMM technique is utilized to solve the problem. This chapter investigates decentralized load balancing in two scenarios: offline EV power allocation for a period with full elasticity for EVs and online EV power allocation with dynamic EV arrivals. By optimizing the power allocation schemes, the global optimization is achieved by local load

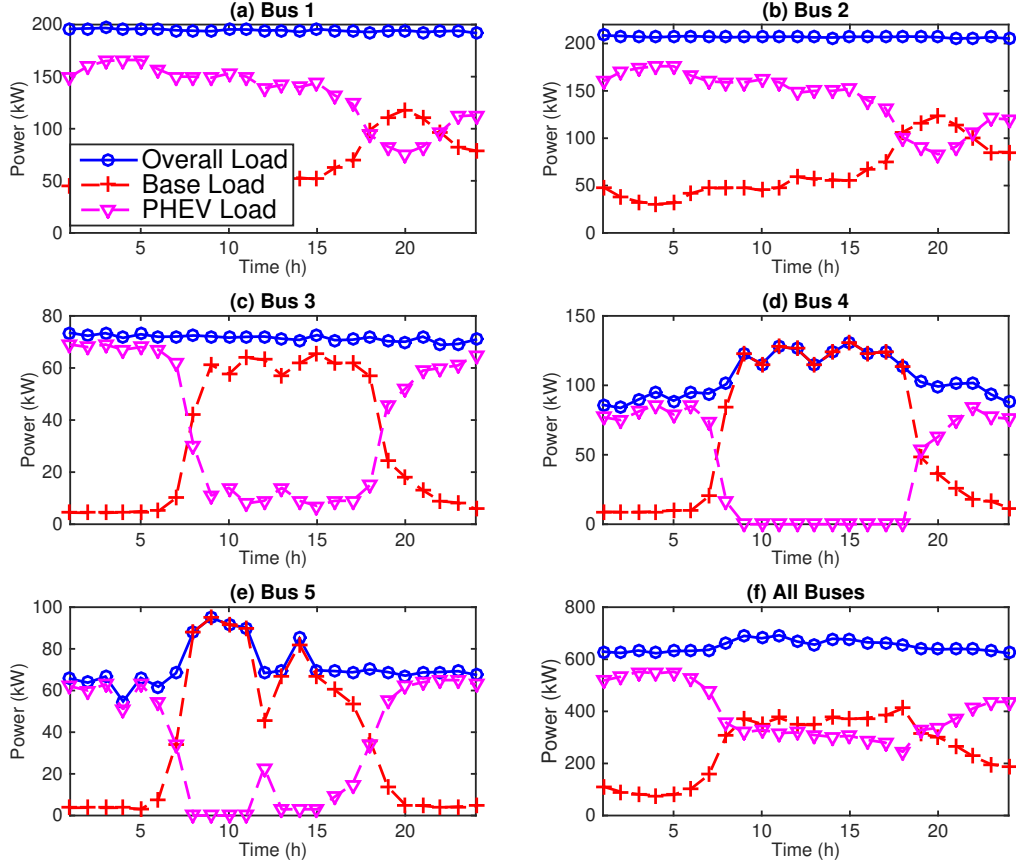


Figure 4.11: The power profile for EV loads and base loads among 5 buses and the whole system by the dynamic EV load allocation scheme in one day.

balancing with minimal global information exchanged among the sub-networks. Meanwhile, the research expands the applications of ADMM tool from canonical optimization problems. The corresponding EV random charging scheme is introduced to schedule EV units according to the optimal power allocation obtained while satisfying charging time requirements of EVs. Finally, the performance on the local and global optimization is presented. The simulation results show that our proposed algorithms are effective methods for balancing the load among sub-networks in a decentralized manner.

Chapter 5

Conclusion and Future Works

5.1 Conclusion

The focus of the study is to investigate efficient load balancing methodologies to minimize the overall power demand fluctuation in the smart grid. In the thesis, two load balancing strategies are discussed: the centralized and the decentralized load balancing strategies. In the research, both static solution and dynamic solution were addressed to response different power circumstances.

For the centralized load balancing problem, the three-dimensional model was formulated into three levels: the entire system, groups, and users. The water-filling approach conducted as the optimization tool to shape the power demand. The basic load allocation scheme is summarized as *BLA* projection to solve the load balancing problem for three levels. The geometric water-filling approach provided the framework of the *BLA* algorithm. The offline elastic load allocation scheme was proposed to achieve load balancing in three-dimension with predicted load information. The corresponding performance was proven to be efficient and optimal for the offline algorithm.

The dynamic centralized load balancing scheme was developed from the offline scheme, and it is applied the geometric water-filling approach in the real-time without the need of the

knowledge of the future load. Two algorithms were proposed to balance the load in a time slot and multiple slots. Online elastic load power allocation (OELPA) algorithm aimed at elastic load power allocation for a time slot, and the computation efficient online approach (EOELPA) aimed to reduce the computation load from OELPA by allocating the elastic loads for multiple future slots. Simulation results were presented to emphasize the overall load fluctuation reduction from the unscheduled case.

Finally, load balancing in sub-networks level was further discussed in the decentralized view. EV loads facilitated load balancing processes among the sub-networks for its mobility and elasticity. The decentralized load balancing strategy was presented by separating the global optimization into the local optimization in sub-networks. Proximal Jacobian ADMM is the optimization tool to realize the decentralized computation. The basic decentralized EV power allocation scheme was proposed for the EV allocation in the period with the load information knowledge, and the dynamic allocation scheme provided the online solution with accommodating the dynamic EV fleets. The simulation results showed the proposed algorithms reducing the power fluctuation locally, and local power allocation achieved the global optimization in a decentralized manner.

The proposed algorithms contribute to the state-of-the-art research on the optimality of load balancing problem in the smart grid. The significant power demand fluctuation reduction offers a stable and efficient power distribution environment. The cooperation between offline and online algorithms enable increasing the reliability of load balancing processes.

5.2 Future Works

On top of the current research work, some research which has not been outlined could be continued as the further consideration. The future works are summarized as follows:

1. The impact of the prediction error should be considered, especially for the offline scenario. The robustness of the proposed algorithms should be improved correspondingly

to accommodate the prediction error.

2. The flexibility of the elastic loads should be considered on top of the research. The characteristics of the elastic load will influence the elasticity of the loads and add more limitations on the power allocation.
3. The queuing length of EVs in the charging station will be considered in the EV power allocation system. The fixed capacity for the charging station will extend the EV waiting time, and the performance on the EV scheduling is affected correspondingly.
4. The approach of load balancing by EV fleetings can be expanded to more sub-networks, rather than within the neighbouring charging stations, by investigating the power loss and the user's intention for EV fleets.

References

- [1] “Ontario demand and market prices,” The Independent Electricity System Operator (IESO), 2016, Available in: <http://www.ieso.ca>.
- [2] J. A. Jardini, C. M. V. Tahan, M. R. Gouvea, S. U. Ahn, and F. M. Figueiredo, “Daily load profiles for residential, commercial and industrial low voltage consumers,” *IEEE Trans. Power Del.*, vol. 15, no. 1, pp. 375–380, Jan 2000.
- [3] X. Fang, S. Misra, G. Xue, and D. Yang, “Smart grid-the new and improved power grid: A survey,” *IEEE Commun. Surveys Tuts.*, vol. 14, no. 4, pp. 944–980, Fourth 2012.
- [4] J. Pan, R. Jain, S. Paul, T. Vu, A. Saifullah, and M. Sha, “An internet of things framework for smart energy in buildings: Designs, prototype, and experiments,” *IEEE Trans. Ind. Informat.*, vol. 2, no. 6, pp. 527–537, Dec 2015.
- [5] M. Shinwari, A. Youssef, and W. Hamouda, “A water-filling based scheduling algorithm for the smart grid,” *IEEE Trans. Smart Grid*, vol. 3, no. 2, pp. 710–719, June 2012.
- [6] K. Spees and L. Lave, “Impacts of responsive load in PJM: load shifting and real time pricing,” *The Energy Journal*, pp. 101–121, 2008.
- [7] T. Logenthiran, D. Srinivasan, and T. Z. Shun, “Demand side management in smart grid using heuristic optimization,” *IEEE Trans. Smart Grid*, vol. 3, no. 3, pp. 1244–1252, Sept 2012.

- [8] Y. Guo, M. Pan, Y. Fang, and P. P. Khargonekar, “Decentralized coordination of energy utilization for residential households in the smart grid,” *IEEE Trans. Smart Grid*, vol. 4, no. 3, pp. 1341–1350, Sept 2013.
- [9] J. S. Vardakas, N. Zorba, and C. V. Verikoukis, “A survey on demand response programs in smart grids: Pricing methods and optimization algorithms,” *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 152–178, 2015.
- [10] “Queensland energy management plan,” Queensland Government, May 2011, Available in: <http://www.cabinet.qld.gov.au>.
- [11] M. Harper, “Review of strategies and technologies for demand-side management on isolated mini-grids,” Mar. 2013, Available in: <http://www.cleanenergyministerial.org>.
- [12] “Managing costs with time-of-use rates,” Ontario Energy Board, 2017, Available in: <https://www.oeb.ca>.
- [13] G. Lacey, G. Putrus, and E. Bentley, “Smart EV charging schedules: supporting the grid and protecting battery life,” *IET Elect. Syst. Transp.*, vol. 7, no. 1, pp. 84–91, 2017.
- [14] G. A. Putrus, P. Suwanapongkarn, D. Johnston, E. C. Bentley, and M. Narayana, “Impact of electric vehicles on power distribution networks,” in *Proc. IEEE Veh. Power Propulsion Conf.*, Sept 2009, pp. 827–831.
- [15] Z. Chen, L. Wu, and Y. Fu, “Real-time price-based demand response management for residential appliances via stochastic optimization and robust optimization,” *IEEE Trans. Smart Grid*, vol. 3, no. 4, pp. 1822–1831, Dec 2012.
- [16] S. Althaher, P. Mancarella, and J. Mutale, “Automated demand response from home energy management system under dynamic pricing and power and comfort constraints,” *IEEE Trans. Smart Grid*, vol. 6, no. 4, pp. 1874–1883, July 2015.

- [17] Y. M. Ding, S. H. Hong, and X. H. Li, “A demand response energy management scheme for industrial facilities in smart grid,” *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2257–2269, Nov 2014.
- [18] J. H. Yoon, R. Baldick, and A. Novoselac, “Dynamic demand response controller based on real-time retail price for residential buildings,” *IEEE Trans. Smart Grid*, vol. 5, no. 1, pp. 121–129, Jan 2014.
- [19] Y. Wang, S. Mao, and R. M. Nelms, “Distributed online algorithm for optimal real-time energy distribution in the smart grid,” *IEEE Internet Things J.*, vol. 1, no. 1, pp. 70–80, Feb 2014.
- [20] L. P. Qian, Y. J. A. Zhang, J. Huang, and Y. Wu, “Demand response management via real-time electricity price control in smart grids,” *IEEE J. Select. Areas Communications*, vol. 31, no. 7, pp. 1268–1280, July 2013.
- [21] P. Yang, G. Tang, and A. Nehorai, “A game-theoretic approach for optimal time-of-use electricity pricing,” *IEEE Trans. Power Syst.*, vol. 28, no. 2, pp. 884–892, May 2013.
- [22] L. Zheng and L. Cai, “A distributed demand response control strategy using lyapunov optimization,” *IEEE Trans. Smart Grid*, vol. 5, no. 4, pp. 2075–2083, July 2014.
- [23] D. Li and S. K. Jayaweera, “Uncertainty modeling and price-based demand response scheme design in smart grid,” *IEEE Sys. J.*, 2014, published on-line at IEEEXplore.
- [24] K. Zhou and L. Cai, “A dynamic water-filling method for real-time HVAC load control based on model predictive control,” *IEEE Trans. Power Syst.*, vol. 30, no. 3, pp. 1405–1414, May 2015.
- [25] Y. Mou, H. Xing, Z. Lin, and M. Fu, “Decentralized optimal demand-side management for phev charging in a smart grid,” *IEEE Trans. Smart Grid*, vol. 6, no. 2, pp. 726–736, March 2015.

- [26] N. Chen, C. W. Tan, and T. Q. S. Quek, “Electric vehicle charging in smart grid: Optimality and valley-filling algorithms,” *IEEE J. Sel. Topics Signal Process*, vol. 8, no. 6, pp. 1073–1083, Dec 2014.
- [27] X. Wang and Q. Liang, “Energy management strategy for plug-in hybrid electric vehicles via bidirectional vehicle-to-grid,” *IEEE Sys. J.*, 2017, published on-line at IEEEXplore.
- [28] R. Yu, J. Ding, W. Zhong, Y. Liu, and S. Xie, “PHEV charging and discharging cooperation in V2G networks: A coalition game approach,” *IEEE Internet Things J.*, vol. 1, no. 6, pp. 578–589, Dec 2014.
- [29] F. Ye, Y. Qian, and R. Q. Hu, “Incentive load scheduling schemes for PHEV battery exchange stations in smart grid,” *IEEE Sys. J.*, 2017, published on-line at IEEEXplore.
- [30] Z. Fan, “A distributed demand response algorithm and its application to PHEV charging in smart grids,” *IEEE Trans. Smart Grid*, vol. 3, no. 3, pp. 1280–1290, Sept 2012.
- [31] Q. Dong, D. Niyato, P. Wang, and Z. Han, “The PHEV charging scheduling and power supply optimization for charging stations,” *IEEE Trans. Vehicular Tech.*, vol. 65, no. 2, pp. 566–580, Feb 2016.
- [32] D. Wu, H. Zeng, C. Lu, and B. Boulet, “Two-stage energy management for office buildings with workplace EV charging and renewable energy,” *IEEE Trans. Transp. Electrification*, vol. 3, no. 1, pp. 225–237, March 2017.
- [33] Y. Kim, J. Kwak, and S. Chong, “Dynamic pricing, scheduling, and energy management for profit maximization in PHEV charging stations,” *IEEE Trans. Vehicular Tech.*, vol. 66, no. 2, pp. 1011–1026, Feb 2017.
- [34] J. Rivera, C. Goebel, and H. A. Jacobsen, “Distributed convex optimization for electric vehicle aggregators,” *IEEE Trans. Smart Grid*, vol. 8, no. 4, pp. 1852–1863, July 2017.

- [35] Z. Li, Q. Guo, H. Sun, and H. Su, “ADMM-based decentralized demand response method in electric vehicle virtual power plant,” in *Proc. IEEE Power Energy Soc. Gen. Meeting*, July 2016, pp. 1–5.
- [36] R. Yu, W. Zhong, S. Xie, C. Yuen, S. Gjessing, and Y. Zhang, “Balancing power demand through EV mobility in Vehicle-to-Grid mobile energy networks,” *IEEE Trans. Ind. Informat.*, vol. 12, no. 1, pp. 79–90, Feb 2016.
- [37] G. Carpinelli, F. Mottola, and D. Proto, “Optimal scheduling of a microgrid with demand response resources,” *IET Generat. Transmiss. Distrib.*, vol. 8, no. 12, pp. 1891–1899, 2014.
- [38] N. Rahbari-Asr and M. Y. Chow, “Cooperative distributed demand management for community charging of PHEV/PEVs based on KKT conditions and consensus networks,” *IEEE Trans. Ind. Informat.*, vol. 10, no. 3, pp. 1907–1916, Aug 2014.
- [39] 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 Trans. Smart Grid*, vol. 6, no. 4, pp. 1795–1805, July 2015.
- [40] M. G. Vay, G. Andersson, and S. Boyd, “Decentralized control of plug-in electric vehicles under driving uncertainty,” in *Proc. IEEE Innov. Smart Grid Technol. Conf.*, Oct 2014, pp. 1–6.
- [41] P. He, L. Zhao, S. Zhou, and Z. Niu, “Water-filling: A geometric approach and its application to solve generalized radio resource allocation problems,” *IEEE Trans. Wireless Communications*, vol. 12, no. 7, pp. 3637–3647, July 2013.
- [42] D. P. Palomar and J. R. Fonollosa, “Practical algorithms for a family of waterfilling solutions,” *IEEE Trans. Signal Processing*, vol. 53, no. 2, pp. 686–695, Feb 2005.

- [43] D. Li, S. K. Jayaweera, and A. Naseri, "Auctioning game based demand response scheduling in smart grid," in *Proc. IEEE OnlineGreenComm*, Sept 2011, pp. 58–63.
- [44] P. He, M. Li, L. Zhao, B. Venkatesh, and H. Li, "Water-filling exact solutions for load balancing of smart power grid systems," *IEEE Trans. Smart Grid*, 2016, published on-line at IEEEXplore.
- [45] M. Li, P. He, and L. Zhao, "Dynamic load balancing applying water-filling approach in smart grid systems," *IEEE Internet Things J.*, vol. 4, no. 1, pp. 247–257, Feb 2017.
- [46] M. Li, P. He, and L. Zhao, "Dynamic elastic load scheduling achieving load balancing for smart grid," in *Proc. IEEE/CIC Int. Conf. Commun. China*, July 2016, pp. 1–6.
- [47] M. Li, L. Zhao, and H. Liang, "An SMDP-based prioritized channel allocation scheme in cognitive enabled vehicular ad hoc networks," *IEEE Trans. Vehicular Tech.*, 2017, published on-line at IEEEXplore.
- [48] M. Baljon, M. Li, and L. Zhao, "SMDP-based resource allocation for wireless networks with energy harvesting constraints," in *IEEE Vehicular Technology Conf.*, Sept 2017, to appear.
- [49] M. Li and L. Zhao, "A decentralized load balancing approach for neighbouring charging stations via EV fleets," in *IEEE Vehicular Technology Conf.*, Sept 2017, to appear.
- [50] N. Kalpana and M. Z. A. Khan, "Fast computation of generalized waterfilling problems," *IEEE Signal Process. Lett.*, vol. 22, no. 11, pp. 1884–1887, Nov 2015.
- [51] K. Naidu, M. Z. A. Khan, and L. Hanzo, "An efficient direct solution of cave-filling problems," *IEEE Trans. Communications*, vol. 64, no. 7, pp. 3064–3077, July 2016.
- [52] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

- [53] W. Deng, M. Lai, Z. Peng, and W. Yin, “Parallel multi-block ADMM with $O(1/k)$ convergence,” *Journal of Scientific Computing*, vol. 71, no. 2, pp. 712–736, 2017.
- [54] T. Erseghe, “Distributed optimal power flow using ADMM,” *IEEE Trans. Power Syst.*, vol. 29, no. 5, pp. 2370–2380, Sept 2014.
- [55] C. Chen, B. He, Y. Ye, and X. Yuan, “The direct extension of ADMM for multi-block convex minimization problems is not necessarily convergent,” *Mathematical Programming*, vol. 155, no. 1-2, pp. 57–79, 2016.
- [56] K. Zhou and L. Cai, “Randomized PHEV charging under distribution grid constraints,” *IEEE Trans. Smart Grid*, vol. 5, no. 2, pp. 879–887, March 2014.