Consumer load modeling and fair mechanisms in the efficient transactive energy market

by

Ahmad Khaled Zarabie

B.S., Kabul University, 2007
M.S., Michigan Technological University, 2012

AN ABSTRACT OF A DISSERTATION

submitted in partial fulfillment of the requirements for the degree

DOCTOR OF PHILOSOPHY

Department of Electrical and Computer Engineering
Carl R. Ice College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2019

Abstract

Two significant and closely related issues pertaining to the grid-constrained transactive distribution system market are investigated in this research. At first, the problem of spatial fairness in the allocation of energy among energy consumers is addressed, where consumer agents that are located at large distances from the substation – in terms of grid layout, are charged at higher rates than those close to it. This phenomenon, arising from the grid's voltage and flow limits is aggravated during demand peaks. Using the Jain's index to quantify fairness, two auction mechanisms are proposed. Both approaches are bilevel, with aggregators acting as interface agents between the consumers and the upstream distribution system operator (DSO). Furthermore, in spite of maximizing social welfare, neither mechanism makes use of the agents' utility functions. The first mechanism is cost-setting, with the DSO determining unit costs. It implements the Jain's index as a second term to the social welfare. Next, a power setting auction mechanism is put forth where the DSO's role is to allocate energy in response to market equilibrium unit costs established at each aggregator from an iterative bidding process among its consumers. The Augmented Lagrangian Multigradient Approach (ALMA), which is based on vector gradient descent, is proposed in this research for implementation at the upper level. The mechanism's lower level comprises of multiple auctions realized by the aggregators. The quasi-concavity of the Jain's index is theoretically established, and it has been shown that ALMA converges to the Pareto front representing tradeoffs between social welfare and fairness. The effectiveness of both mechanisms is established through simulations carried out using a modified IEEE 37-bus system platform.

The issue of extracting patterns of energy usage from time series energy use profiles of individual consumers is the focus of the second phase of this research. Two novel approaches for non-intrusive load disaggregation based on non-negative matrix factorization (NMF), are proposed. Both algorithms distinguish between fixed and shiftable load classes, with the latter being characterized by binary OFF and ON cycles. Fixed loads are represented as linear combinations of a set of basis vectors that are learned by NMF. One approach imposes L0 normed constraints on each shiftable load using a new method called binary load decomposition. The other approach models shiftable loads as Gaussian mixture models (GMM), therefore using expectation-maximization for unsupervised learning. This hybrid NMF-GMM algorithm enjoys the theoretical advantage of being interpretable as a maximum-likelihood procedure within a probabilistic

framework. Numerical studies with real load profiles demonstrate that both algorithms can effectively disaggregate total loads into energy used by individual appliances. Using disaggregated loads, a maximum-margin regression approach to derive more elaborate, temperature-dependent utility functions of the consumers, is proposed. The research concludes by identifying the various ways gleaning such information can lead to more effective auction mechanisms for multi-period operation.

Consumer load modeling and fair mechanisms in the efficient transactive energy market

by

Ahmad Khaled Zarabie

B.S., Kabul University, 2007
M.S., Michigan Technological University, 2012

A DISSERTATION

submitted in partial fulfillment of the requirements for the degree

DOCTOR OF PHILOSOPHY

Department of Electrical and Computer Engineering
Carl R. Ice College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2019

Approved by:

Major Professor
Dr. Sanjoy Das

Copyright

Abstract

Two significant and closely related issues pertaining to the grid-constrained transactive distribution system market, are investigated in this research. At first, the problem of spatial fairness in the allocation of energy among energy consumers is addressed, where consumer agents that are located are large distances from the substation – in terms of grid layout, are charged at higher rates than those close to it. This phenomenon, arising from the grid's voltage and flow limits; is aggravated during demand peaks. Using the Jain's index to quantify fairness, two auction mechanisms are proposed. Both approaches are bilevel, with aggregators acting as interface agents between the consumers and the upstream distribution system operator (DSO). Furthermore, in spite of maximizing social welfare, neither mechanism makes use of the agents' utility functions. The first mechanism is cost-setting, with the DSO determining unit costs. It implements the Jain's index as a second term to the social welfare. Next, a power setting auction mechanism is put forth where the DSO's role is to allocate energy in response to market equilibrium unit costs established at each aggregator from an iterative bidding process among its consumers. The Augmented Lagrangian Multigradient Approach (ALMA), which is based on vector gradient descent, is proposed in this research for implementation at the upper level. The mechanism's lower level comprises of multiple auctions realized by the aggregators. The quasi-concavity of the Jain's index is theoretically established, and it has been shown that ALMA converges to the Pareto front representing tradeoffs between social welfare and fairness. The effectiveness of both mechanisms is established through simulations carried out using a modified IEEE 37-bus system platform.

The issue of extracting patterns of energy usage from time series energy use profiles of individual consumers is the focus of the second phase of this research. Two novel approaches for non-intrusive load disaggregation based on non-negative matrix factorization (NMF), are proposed. Both algorithms distinguish between fixed and shiftable load classes, with the latter being characterized by binary OFF and ON cycles. Fixed loads are represented as linear combinations of a set of basis vectors that are learned by NMF. One approach imposes L0 normed constraints on each shiftable load using a new method called binary load decomposition. The other approach models shiftable loads as Gaussian mixture models (GMM), therefore using expectation-maximization for unsupervised learning. This hybrid NMF-GMM algorithm enjoys the theoretical advantage of being interpretable as a maximum-likelihood procedure within a probabilistic

framework. Numerical studies with real load profiles demonstrate that both algorithms can effectively disaggregate total loads into energy used by individual appliances. Using disaggregated loads, a maximum-margin regression approach to derive more elaborate, temperature-dependent utility functions of the consumers, is proposed. The research concludes by identifying the various ways gleaning such information can lead to more effective auction mechanisms for multi-period operation.

Table of Contents

List of Figures xi
List of Tables xiii
Acknowledgements xiv
Dedication xv
Chapter 1 - Introduction 1
1.1 Literature Review 2
1.2 Contributions 7
1.2.1 Fairness-Regularized DLMP-Based Cost-setting Mechanism in Transactive Energy Market 7
1.2.2 Pareto-Optimal Energy Allocation Mechanism in Transactive Energy Market 9
1.2.3 Load Disaggregation and Energy Consumer Modeling 10
Chapter 2 - Fairness-Regularized DLMP-Based Cost-setting Mechanism in Transactive Energy Market 12
2.1 Introduction 12
2.2 Framework 12
2.2.1 Prosumer Agent 13
2.2.2 Aggregator 15
2.2.3 DSO 15
2.3 Mathematical Model 16
2.3.1 Linearized AC Power Flow 16
2.3.2 Jain's Fairness Index 18
2.3.3 DSO Mechanism 20
2.4 Simulation Results 23
2.5 Conclusion 27
Chapter 3 - Pareto-Optimal Energy Allocation Mechanism in Transactive Energy Market 30
3.1 Introduction 30
3.2 Framework 31
3.2.1 Aggregator Mechanism 31
3.2.2 Constraints 35

3.2.3 Jain's Index of Fairness 35
3.2.4 Constrained Vector Optimization Problem Formulation 36
3.3 Proposed Approach 37
3.3.1 Theoretical Background 37
3.3.2 Augmented Lagrangian Multi-Gradient Ascent 39
3.3.3 DSO Level Pareto-Optimality 42
3.3.4 DSO Algorithm 44
3.4 Simulation Results 46
3.5 Conclusion 49
Chapter 4 - An L0-Norm Constrained Non-Negative Matrix Factorization Algorithm for the Simultaneous Disaggregation of Fixed and Shiftable Loads 51
4.1 Introduction 51
4.2 Framework 52
4.2.1 Load Models 52
4.2.2 Objective 54
4.3 Proposed Approach 54
4.3.1 Probabilistic Framework 54
4.3.2 Multiplicative Update 56
4.3.3. Sparsity Constrained Binary Updates 57
4.3.4 NMF Algorithm 62
4.4 Simulation Results 63
4.5 Conclusion 69
Chapter 5 - A Data Driven Approach for Energy Consumer Modeling 70
5.1 Introduction 70
5.2 Overall Approach 72
5.2.1 Load Model 72
5.2.1.1 Fixed Load 73
5.2.1.2 Shiftable Loads 73
5.2.3 Algorithmic Framework 75
5.3 Load Disaggregation 76
5.3.1 Gaussian Mixture Model 76

5.3.2 Non-Negative Matrix Factorization 77
5.4 Parameter Estimation 78
5.4.1 Constraints 78
5.4. 2 Objective 79
5.4.3 Quadratic Programming 79
5.5 Simulation Results 80
5.5.1 Data Processing 80
5.5.2 Load Disaggregation 81
5.6 Conclusion 87
Chapter 6 - Conclusion 89
References 91
Appendix A - Nomenclature 99
Nomenclature for Chapter 2 99
Nomenclature for Chapter 3 100
Appendix B - Linearized Constraints 101

List of Figures

Figure 2.1: Schematic of the functional components of the grid and the bi-level flow of information between them. 13
Figure 2.2: Utility of a prosumer as a function of energy consumed. 14
Figure 2.3: Distribution system radial branch model. 16
Figure 2.4: IEEE 37 bus system with aggregators indexed A1 – A17. 23
Figure 2.5: Results of Scenario-I showing unit costs and allocated power to each aggregator (top) and the DLMP components (bottom). 24
Figure 2.6: Results of Scenario-II showing unit costs and allocated power to each aggregator (top) and the DLMP components (bottom). 25
Figure 2.7: Results of Scenario-III showing unit costs and allocated power to each aggregator (top) and the DLMP components (bottom). 26
Figure 2.8: Total welfare and price of fairness vs. Jain's index 27
Figure 3.1: Schematic of market-driven bilevel mechanism 31
Figure 3.2: Typical quasiconcave utility function of agent $i \in \mathcal{G}k$ 32
Figure 3.3: Convergence towards fixed point of aggregator auction 33
Figure 3.4: Feasible regions of ω. 40
Figure 3.5: Common feasible ascent direction 42
Figure 3.6: IEEE 37-bus system used as simulation platform 46
Figure 3.7: Aggregator power allocations and unit costs 47
Figure 3.8: Welfare and Jain's index (top), inner product of gradients of welfare and fairness (bottom) vs. iteration 48
Figure 3.9: Jain's index vs. welfare 48
Figure 4.1: . Schematic of NMF factorization with separate basis and weight matrices for the fixed and shiftable loads. 53
Figure 4.2: Illustrative usage profile of a shiftable load showing a maximum of Lj duty cycles, with each such cycle being a rectangular pulse of uniform peak pj. 54
Figure 4.3: Disaggregated shiftable loads (top) and aggregate loads (bottom) for Day-1. 64
Figure 4.4: Disaggregated shiftable loads for Day-1. 66
Figure 4.5: Disaggregated shiftable loads (top) and aggregate loads (bottom) for Day-2. 67

Figure 4.6: Disaggregated shiftable loads for Day-2. 68
Figure 5.1: Schematic showing typical profiles of fixed (top) and shiftable (bottom) loads. 73
Figure 5.2: Isoquant for a Cobb-Douglas preference function (black). For comparison, isoquants for perfect substitutes (blue) and perfect complements (red) are also shown in dashed lines. 75
Figure 5.3: Steps used in the proposed approach. Arrows depict flow of information 76
Figure 5.4: Shiftable load actual (top) and computed (bottom) for User-1 82
Figure 5.5: Fixed load actual (top) and computed (bottom) for User-1..... 83
Figure 5.6: Aggregate load actual (top) and computed (bottom) for User-1. 83
Figure 5.7: Shiftable load actual (top) and computed (bottom) for User-2. 84
Figure 5.8: Fixed load actual (top) and computed (bottom) for User-2..... 84
Figure 5.9: Aggregate load actual (top) and computed (bottom) for User-2. 85
Figure 5.10: Actual vs. computed values of parameter a..... 85
Figure 5.11: Actual vs. computed values of parameter d. 86
Figure 5.12: Actual vs. computed values of parameter $x0$. 86
Figure 5.13: Utility curves(top), cost curves (bottom) 87

List of Tables

Table 4.1 Duty Cycle Parameters 64
Table 5.1: Scaler utility parameter 81
Table 5.2: Load model parameters 82

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Dr. Sanjoy Das, for his patience, unwavering support, and consistent encouragement. His guidance, knowledge, and genuine care is the driving force behind this achievement. Dr. Das is a great professor and I am grateful to be his student. He has always been a caring and understanding mentor.

My genuine gratitude to my doctoral committee members Dr. Anil Pahwa, Dr. Hongyu Wu, Dr. Scott DeLoach, and Dr. Christine Aikens for their support and valuable feedback to improve the quality of my research.

I would like to thank Dr. Mohmmad Nazif Faqiry for his friendship, guidance and collaboration. Special thanks to Ms. Sahar Lashkarbolooki for her encouragement, support and collaboration.

My heartfelt appreciation to my lovely siblings, Wahida, Nooria, Madeha, Baset and Belal Zarabie and my esteemed uncles, late Abdul Wahab Zarabie and Abdullah Zarabie, my dear cousins, Dr. Abdul Salam Rahmani, Dr. Abdul Latif Rahmani and Dr. Wahidullah Zarabie for their unquestioned support, care and encouragement which helped me to accomplish this major achievement.

I would like to truly thank my beautiful and caring wife, Sweeta Zarabie, and our lovely children, Marwa and Yama Zarabie, for their unconditioned love, patience, and sacrifice during my education. These past several years have not been an easy ride, both academically and personally. I truly thank you for sticking by my side, even when I was irritable and depressed.

This work was supported by National Science Foundation through award No. CNS-1544705.

Dedication

I dedicate this thesis to my beloved mother Hamida Zarabie and my dear uncle Dr. Asadullah Zarabie.

Chapter 1 - Introduction

Traditionally, electricity customers had little role in the electricity market. The flow of energy was unidirectional, with residential units – the customers of energy, acting merely as passive agents who were charged at a flat rate. However, due to the increased use of renewable energy and distributed energy resources (DERs), this situation is witnessing a paradigm shift. As *prosumers*, who not only use the resource, but also produce their own locally generated energy, and furthermore, are willing to trade their surplus in an energy marketplace, customers will participate proactively in the energy market, as key stakeholders.

The exchange of electrical energy between the grid and the prosumers requires efficient market-based approaches. These approaches are termed as *transactive energy*. Transactive energy as defined by the U.S. Department of Energy in the GridWise report, 2015 as, *"A system of economic and control mechanisms that allows the dynamic balance of supply and demand across the entire electrical infrastructure using value as a key operational parameter"*.

The Transactive energy market approaches lay the ground for a setting in which the major electricity producer and the electricity customers can work closely together and match and balance their supply and demand more efficiently. The Transactive energy participants as well as the society can gain potential benefits from it. The rapid advances in communication technologies, coupled with the emergence of the field of distributed artificial intelligence, together will lead to significant improvements in efficiency, resiliency, and reliability of the electrical grid, with such a transactive energy framework. However, it is important to study and understand the overall implications and impacts of the Transactive energy approaches and utilize the right control tools to manage and implement it.

There are four key challenges towards realizing this goal, which are outlined below.

- Market efficiency: Transactive energy should lead to growth in the social welfare. With customers acting as selfish agents with the aim of maximizing their own benefits, not only from utilizing energy but also in terms of monetary gains, it is imperative to introduce efficient distributed mechanisms for two-way exchange of electricity in the energy market.
- Customer motivation: Customers should be willing to assume their new roles, which would occur as long as their exists in them, an intrinsic sense of fairness in the manner in which the

resource is shared. The Transactive energy mechanisms must also be based on the limited amount of information that the customers are willing to share publicly.

- Prosumer model: Customer participation would be enabled only through the use of automation, which would require precise models of the choices that they exercise in the market.
- Physical grid constraints: Unlike the traditional marketplace, physical constraints of the energy grid will impose additional restrictions on the exchange of energy, which will have to be addressed by this transactive framework.

1.1 Literature Review

The rapid proliferation of distributed energy resources (DERs) in the energy grid necessitates the need for the design of efficient transactive distribution system markets (DSMs). Pricing mechanisms that are compatible with the physical structure of the distribution grid and take operation limits into account, are being proposed in the literature [1], [2], [3], [4], [5], [6]. Distribution locational marginal pricing (DLMP) as an effective means to establish the price of electricity in transactive DSMs has received significant research attention [2]-[6].

The latest research on DLMP-based pricing decompose locational prices to its energy, losses, voltage violation, and congestion components, [4], [5]. An inherent drawback of this method is the spatial variations in the resulting DLMP. When the grid is under stress (e.g., due to a line congestion or a node voltage hitting its operation limit), the DLMP-based pricing intrinsically charges distant nodes at higher rates than those closer to the substation. In particular, the effect of location in DLMP is substantially high if an extreme node violates the voltage constraint due to higher voltage drop.

DLMP-based pricing methods and its DSM models either use DCOPF or a variant of AC optimal power flow (ACOPF) to set grid and operation limit constraints. A few papers on DCOPF-based DLMP formulations have appeared [2], [7], and [8], [9]. References [2] and [7], propose DLMP-based methods through quadratic programming and chance-constrained mixed integer programming that use DCOPF to define line congestion and alleviate it through dynamic tariffs. The work in [8] proposes two benchmark pricing methodologies, namely DLMP and iterative DLMP (iDLMP), for a congestion free energy management by buildings providing flexible demand. Aggregators are assumed to have contracts with flexible buildings to decide their reserve

and energy schedules by interacting with the DSO in a cost optimal manner to avoid congestion in the distribution grid. An augmented version of DCOPF to include losses in a DLMP-based pricing mechanism with hedging rights for flexible load in distribution grid has been used in [9]. Unfortunately, due to lower x/r ratio in the distribution system, DCOPF-based DLMP has been shown to introduce significant errors [3]. Moreover, these techniques lack certain key features such as losses, voltage deviations, and reactive power flows essential in transactive DSM.

ACOPF-based DLMP formulations to determine one or more of its constituents such as energy, loss, voltage violation, and congestion prices are investigated in [3], [4], [5]. In [3], a novel linearized power flow (LPF) method is proposed. In this paper, the real and reactive energy and the loss components of the DLMP are derived. In [4], a DSM model with DLMP clearing has been proposed to manage congestion and provide voltage support. The paper uses a mixed-integer second-order-cone programming to model ACOPF and determines binary variables such as feeder configuration status and tap locations of shunt capacitors and transformers. Similarly, in [5], the authors use a trust-region based solution methodology to obtain DLMP and its constituents through a first-order approximation of the AC power flow manifold model.

The issue of fairness in pricing has not received due attention in the above research. Fairness considerations in other forms of pricing have begun to be addressed recently. Several papers, [10], [11], [12], [13] make use of the Shapley value, a concept borrowed from coalitional game theory, to accomplish fairness. In [10] prosumers' fair hourly billings is achieved depending on how the DR meet system objectives. The price of anarchy, which is the deviation of the Nash equilibrium operating point from the optimal has been used to incorporate fairness in hourly billings to prosumers in [14], and as a basis for comparing two models of demand side management in terms of fairness in [15]. A method to determine fair energy costs to consumers based on their contribution to minimize overall system costs is proposed in [11]. Several fairness criteria based on emission minimization, minimize peak-to-average ratio, etc. have been proposed in [16]. In [12], a pricing during direct trade among prosumers energy is proposed using the Shapley value. A fairness-based criterion is proposed in [17] to share the cost savings in a coalition of prosumers equipped with renewable energy sources and energy storage systems.

The aim of these approaches is to reward users that consume energy during more desirable time intervals and conversely, to penalize those that demand energy during undesirable periods. We term this aspect of fairness as temporal fairness. The expectation of temporal fairness is

typically to redirect the grid's operation towards more feasible operating regions. There is another aspect of fairness in addition to temporal fairness that is described below.

Energy consumption by a group of users in one area of the grid affects how the other users in the grid are priced. For instance, when consumers in a node that is positioned close to the substation transformer draw a disproportionately large amount of energy, the amounts that those further downstream can obtain from the grid is stymied. It is this relative advantage or drawback of the end users, which is based on their locations that this research proposes to mitigate. It does so by incorporating another form of fairness, which we term as spatial fairness.

A few studies use other mechanisms that consider the loads' locations to fairly redistribute prices [9], [18]. In [9], the authors use hedging rights, a concept borrowed from the wholesale market, to mitigate the undesirable effects of physical-grid-based DLMPs to the aggregators that provide load flexibility service and achieve fairer price redistribution. Day-ahead DLMPs are computed based on an augmented version of DCOPF. Monthly contracted hedging rights are then used to hedge the risk of surcharges arising from the DLMP's congestion/loss components to the aggregators. In [18], the authors use an indirect mechanism based on load location and their marginal contribution to the losses and voltage deviation to attain fair billing. Although these mechanisms mitigate the inherent spatial unfairness in DLMP pricing, they achieve fairer pricing indirectly by means of a secondary corrective step.

Vector optimization are optimization approaches with vector objective functions. For decades, the simultaneous optimization of more than one objective function, has been implemented by means of multi-objective evolutionary algorithms such as genetic algorithms and particle swarm optimization (cf. [19], [20]). They have been applied to a plethora of problems in the energy market [21], [22], [23], [24], [25]. Multi-objective evolutionary algorithms are well equipped to handle constraints by either repairing infeasible solutions or simply rejecting them. Unfortunately as they incorporate stochastic operations, multi-objective evolutionary algorithms have to evaluate a large number of poor solutions (in terms of the objective functions) before converging to a Pareto-optimal set. As population-based approaches, evolutionary optimization cannot directly be applied to improve within a few quick steps any existing solution that is already close to Pareto optimality. Lastly, theoretical convergence guarantees of this class of algorithms are only of an indirect nature that approximate the algorithmic processes as discrete Markov chains. Lastly,

evolutionary methods tend to be applied in situations without ascertaining the presence of a large number of local optima a priori, where simpler methods would have sufficed.

Deterministic vector optimization is a newer alternative to evolutionary algorithms. Normalized boundary intersection (NBI) is an indirect method of scalarization. It identifies the ideal solution in the objective function space from known theoretical bounds. Thereafter NBI directs the search towards the ideal point by means of conventional scalar optimization techniques. In [26], which applies NBI, the vector objective comprises of the scalar objectives of all aggregators. NBI is proposed as a method to minimize a vector of uncertainties in pricing in [27]. Unfortunately, NBI is prone to yielding solutions that are not in the Pareto front (i.e. the image manifold of the Pareto-optimal set). Conversely there are some regions in the Pareto front that are inaccessible to NBI [28].

Multi-gradient algorithms are a class of nascent algorithms that extend the steepest ascent method to vector objectives, some of them subsequently extended to quasi-Newton and second order methods. The early work in [29] defines a feasible ascent direction in terms of Hessian approximation. The multi-gradient descent approach (MGDA) in [30] uses a geometric definition of a feasible ascent direction. As MGDA is, to the best of the authors' knowledge, the only multi-gradient algorithm to have been adopted for any significant application domain [31], [32], it forms the basis of the approach proposed here. Unfortunately, MGDA is not equipped to handle constraints [33]. In [34], and more recently in [35], penalty function approaches are proposed to handle constraints. A very recent approach for constrained vector optimization (CVOP) has been proposed in [36] that extends Zoutendijk's method to handle active constraints.

It must be noted that there are numerous other approaches through which vector objectives can be handled, such as optimizing the weighted sums of objectives, lexicographic ordering, or elastic constraint methods [37]. The above discussion was confined only to the major classes of approaches that have found energy grid applications along with vector gradient ascent, which is relevant here.

Recent advancements in machine learning have made it possible to disaggregate residential loads obtained from smart meters into smaller components. Some algorithms for load disaggregation rely on supervised or semi-supervised learning, which require a set of training data where the individual loads are known a priori [38]. Deep neural networks are a common choice of such methods [39], [40], [41], [42]. Unfortunately, these methods are not suitable for non-intrusive

load monitoring where the only available data is in the form of smart meter readings, even during the training phase.

Hidden Markov models (HMM) are a popular choice of modeling individual appliances. In all these methods, individual appliances are represented using two or more states, with the transition and/or emission probabilities obtained through a number of learning methods, usually derived from expectation-maximization (EM). In [43], the total number of appliances are determined using *K*-means clustering. An improved method proposed in [44] that models each appliance using a factorial HMM, each of which contains more than a single HMM. Other extensions include used to represent appliances include explicit HMMs [45] and hierarchical HMMs [46] for load modeling.

The EM algorithm is also used to train Gaussian mixture models (GMM) [47]. GMMs are based on probability theory, that usually apply the maximum likelihood criterion for appliance classification. These approaches assume the presence of a set of latent variables z_j, where j is an index, with only one being 'active' at each time. The output follows a Gaussian probability distribution that is uniquely identified by the active latent variable. The algorithm proposed in [47] is based on GMMs. The approach uses the Dempster-Shaffer theory for appliance classification.

Non-negative matrix factorization (NMF) is widely used machine learning approach for energy disaggregation. The classical NMF algorithm decomposes an input data matrix $\mathbf{X}$ whose columns are sample vectors, into two factors, $\mathbf{W}$ and $\mathbf{H}$, so that their product equals $\mathbf{X}$. Usually, $\mathbf{X}$ has a very large number of columns, which are independent samples. The columns of $\mathbf{W}$ serve as basis vectors so that each sample $\mathbf{x}(n), n \in \mathcal{N}$, which are columns of $\mathbf{X}$ can be represented as a weighted combination of the bases, with the non-negative weights being the corresponding column vectors $\mathbf{h}(n)$ of $\mathbf{H}$. NFM has been used for energy disaggregation of HVAC load components in an industrial building and in a smart home setting [48]. Another method has been proposed in [49] , [50] to impose L_0 constraints, which uses a softmax distribution for the elements in $\mathbf{H}$ to assign weights to them in such a manner that those with higher values are likelier to improve the objective function. Semi-supervised NMF using prior knowledge of the usage time profiles of individual appliances has been proposed in [38]. NMF has been applied for data over larger periods in time to retrieve seasonal trends in usage profiles in [51].

Several other approaches that are tailor-made for energy disaggregation have been proposed. Among them include [52] that uses the fuzzy c-means clustering to identify the number

of appliances. Quadratic programming has been used in [53]. A computationally efficient additive neural network is used in [54], with a specialized training algorithm called cogent confabulation. The approach in [54] relies on OFF-ON transitions of appliances to detect individual appliances. In this approach, bagging – a method of combining classifiers into ensembles, has been adopted for improved classification of appliances from load signatures. Two novel algorithms based on spectral graph theory has been applied for disaggregation in [55] whereas fuzzy logic has been adopted in [56].

In [57], generation side utilities are modeled as piecewise linear functions so that linear programming can be applied for energy auction. A linear utility function is also used in [58] for individual appliances, with customers trying to maximize the sum of all such utilities. However, it should be noted that in linearization are not suitable in many other applications as they do not encapsulate the dependence of utility to price changes. For this reason, recent research proposals routinely use saturating nonlinearities as utilities. In [59] customers' behavior based utility functions are modeled in this manner. In the double auction mechanism described in [60] where participants' bidding strategies are determined from nonlinear utilities of individual goods. Similarly, in [61], nonlinear functions have been used to quantify the utility of a potential user from pricing and QoS.

Recently, logarithmic functions are commonly used to model utilities. The double auctions for signal-to-interference plus noise ratios as well as for power allocation in [60] make use of such utilities. Portfolio optimization using power-log utilities are taken up in [62]. The approach in [63] uses risk averse and risk neutral log-concave utility functions. In [64], concave as well as non-concave utilities have been used, with logarithmic functions to represent concave utilities and sigmoidal nonlinearities for the latter. Logarithmic utilities are also used to quantify the benefit derived from operating shiftable appliances.

1.2 Contributions

1.2.1 Fairness-Regularized DLMP-Based Cost-setting Mechanism in Transactive Energy Market

This research is presented in chapter 2.

The key features of the approach proposed in this research are summarized below.

(i) The proposed DLMP pricing mechanism addresses spatial fairness. This is accomplished by incorporating a regularization term in the system level objective function (addressed below). The Jain's index of fairness has been used for this purpose. Jain's index is an instance of a general class of fairness criteria that possess desirable features [65], [66] that render them particularly well-suited for user-centric resource allocation applications [67]. Furthermore, unlike its use as an evaluative tool [68] or as a system constraint [69], the proposed approach successfully includes Jain's index directly within the optimization algorithm, as well as in the DLMP pricing.

(ii) The proposed framework accommodates physical constraints of the grid. It uses a linearized power flow method. Similar approaches have been used elsewhere [1], [5], [70], [71]. Linearizing the power flow in this research, which is directly based on [3], not only helps in simplifying the underlying computations, but also allows the components of the DLMP to be readily available.

(iii) Prosumers in this framework are not required to reveal private information despite having their own DGs and individual utility functions. The only information exchange taking place between individual prosumers and the rest of the grid is limited to placing power demand bids in response to unit energy costs, i.e. DLMPS.

(iv) If the regularization term is neglected, the proposed method maximizes the social welfare of all prosumers in the grid, i.e. the sum of all their utilities. It should be noted that this task is accomplished at the DSO level despite its lack of access to the nonlinear utility functions of individual prosumers. The proof that the maximum social welfare is attained appears in a preliminary version of one aspect of this research [72], which neither considers Jain's index or any other fairness measure, nor addresses the constrained optimization algorithm used here.

(v) The proposed approach is a bi-level mechanism, with the DSO and the prosumers aiming to maximize different objectives. Prosumers are modeled as selfish agents that aim to maximize their individual payoffs, i.e. the difference between their utilities from consuming energy and the cost of procuring it from the grid (with

negative demands indicating supply). Aggregators act as the interface between prosumers and the grid.

(vi) The underlying optimization is based on dual decomposition. It uses the augmented Lagrangian method [73] at the DSO level for social welfare maximization, obviating the need for off-the-shelf solvers

1.2.2 Pareto-Optimal Energy Allocation Mechanism in Transactive Energy Market

This research is presented in chapter 3. This research proposes a novel approach to obtain Pareto optimal energy allocations representative of the tradeoff between efficiency and fairness, where fairness is quantified in terms of the Jain's index. At first, necessary and sufficient conditions for Pareto optimality are formally established. The overall problem is formulated in terms of a constrained vector optimization problem (CVOP), with the constraints comprising of the physical grid's voltage deviation and power flow limits, as well as cost and energy balance conditions. This vector objective framework allows a two-stage optimization algorithm to obtain tradeoff energy allocations that compromise some efficiency for more fair allocation.

The major contributions of the proposed research are categorized below.

(i) The proposed ALMA relies on the recently proposed MGDA, which has so far only found limited applications. Furthermore, ALMA is a novel scheme that couples the well-known augmented Lagrangian method, which is intended for constrained scalar optimization, with an enhanced version of MGDA that can now handle constraints. To the best of the authors' knowledge, such an approach has not been used elsewhere in constrained vector optimization. Moreover, ALMA is a general-purpose approach for CVOP that can readily be adopted to similar engineering and other domains.

(ii) Although there is no dearth of literature on multi-objective optimization in energy systems, again to the best of the authors' knowledge, this research is the first to introduce to the energy systems community, an emergent class of vector optimization methods along with its accompanying mathematical underpinnings.

(iii) At the same time, the energy market offers the opportunity for ALMA's performance to be evaluated for large scale optimization, under the presence of a

large number of constraints, that when put together outnumber the number of decision variables.

(iv) ALMA is implemented within a power-setting mechanism, thereby inheriting all the advantages proffered by cost-setting mechanism.

(v) The aggregator level auction used here is an improved version of that in [1], [73]. Agents no longer have to declare their intended roles as buyers or sellers beforehand; they can switch roles at any step based on changing energy costs during the auction.

(vi) As ALMA is built-in to this framework, off-the-shelf solvers are no longer required for optimization. This allows direct access to all quantities involved, including dual variables.

(vii) It is shown that Jain's fairness index is quasiconcave everywhere in the design space. Thereby, it extends previous results that showed its concavity only in the first orthant [72]. This is a significant result for any application that uses the index within an optimization procedure. Moreover, it is shown that the output of ALMA is Pareto-optimal as long as the objective functions are quasiconcave. Hence, ALMA is not restricted to purely concave utilities as in [5], [9], [71], [73],[72], [74].

(viii) To the extent of the authors' knowledge, this is the first attempt to seek Pareto-optimal efficiency-fairness tradeoffs with the Jain's index paired with the generic, widely accepted measure of welfare, instead of more synthetic measures of utilities that are specific to the Jain's index, i.e. α-fair utility function [65], [67].

This research is presented in chapter 3.

1.2.3 Load Disaggregation and Energy Consumer Modeling

The novelties of the research that this chapter entails are in the following directions. These are outlined below.

(i) A fully non-intrusive approach for load disaggregation is proposed. This is an improvement not only over supervised learning approaches but also those that use unsupervised learning such as GMMs, HMMs, or NMF, most of which require

some form of prior information about the consumption patterns of individual appliances. Therefore, the proposed approach can be entirely trained in real time after deployment.

(ii) Disaggregation in this research is accomplished with a hybrid algorithm that harnesses the advantages of GMMs and NMF. GMMs are most effective in learning the load patterns of binary OFF-ON shiftable appliances as well as those operating at a few discrete levels, whereas NMF approaches are suitable to iteratively obtain basis sets to represent fixed loads whose energy consumption levels cannot be discretized readily.

(iii) A semi-parametric consumer utility model has been proposed in this research, this represents the appropriate tradeoff between parametric and non-parametric models. For many applications, it suffices to distinguish only between fixed and cost dependent components of the load.

(iv) Additionally, the present utility model considers temperature dependence of energy consumption.

(v) Parameter estimation of the consumer utility model has been formulated in terms of L_2-norm maximum margin regression.

(vi) The use of the utility model as a means to validate the performance of the disaggregation algorithm has been proposed.

This research is presented in chapters 4 and 5.

Chapter 2 - Fairness-Regularized DLMP-Based Cost-setting Mechanism in Transactive Energy Market

Distribution locational marginal pricing (DLMP) can adversely affect users in a grid-constrained transactive distribution system market (DSM) that are at larger distances from the substation, thereby requiring longer paths to connect to it. When the grid operates closer to its physical limits in terms of line capacities and voltage deviations, these users are more likely to cause grid violations than others in the vicinity of the substation. Conversely, increased energy consumption by users near the substation can choke off supply to those at the grid's extremities. This research describes a novel mechanism to charge users in a more equitable manner, by regularizing the distribution system operator (DSO)'s social welfare objective function with the Jain's index of fairness. The overall problem entails the maximization of the regularized objective within a set of linear constraints that ensure that the grid's physical limits are not violated. Dual decomposition is applied to the constrained optimization problem. The dual variables and unit costs are incrementally updated by the DSO using the augmented Lagrangian method (ALM). Simulation results confirm the effectiveness of the proposed approach.

2.1 Introduction

In this chapter, we propose a novel method that addresses the issue of fairness in DLMP-based DSM. We propose a fairness-regularized mechanism that can be implemented by the distribution system operator (DSO). An iterative gradient descent algorithm based on dual decomposition maximizes the global social welfare of the grid, but with spatially driven discrepancies in how the prosumers within the aggregators are charged. Incorporating fairness into the algorithm's objective has a demonstrably equitable effect on the unit energy costs.

A list of abbreviations for this chapter is provided in Appendix A.

2.2 Framework

The overall schematic of the bi-level mechanism is depicted in Figure 2.1. At the upper level is the DSO which acts as the intermediary between the grid and the wholesale market. The DSO

possesses physical information pertaining to the distribution grid and exchanges unit cost and power demand signals from each aggregator, $k \in \mathcal{A}$. Aggregators are located at some nodes of the grid, which follows a tree structured layout. Only a subset $\mathcal{A}$ of $\mathcal{N}$ contain aggregators. Each aggregator k contains a set $\mathcal{G}_k$ of prosumers within a physical neighborhood. The information flow between an aggregator and its prosumers again pertain to unit costs and demands.

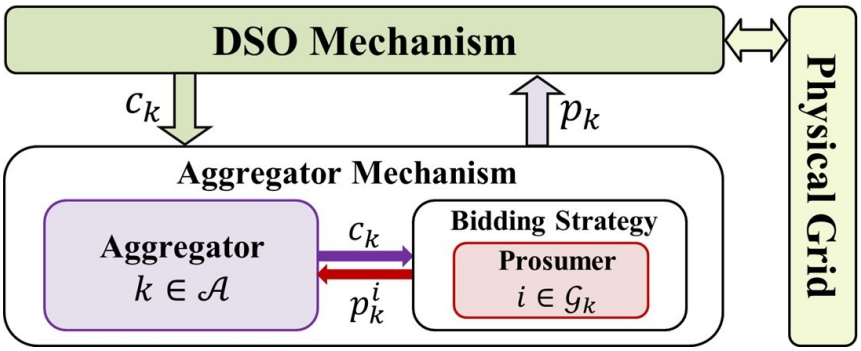

Figure 2.1: Schematic of the functional components of the grid and the bi-level flow of information between them.

2.2.1 Prosumer Agent

In a transactive DSM, contemporary retail customers are key stakeholders – prosumer agents in this framework. Each prosumer $i \in \mathcal{G}_k$ incorporates a utility function that may be construed as a measure of the satisfaction it derives from using a certain amount of energy. The utility $u_k^i: \mathbb{R}_+ \to \mathbb{R}_+$ of prosumer i is a strictly concave and increasing function that is continuous and differentiable. Prosumer utilities have been modeled as logarithmic functions in the following manner,

$$u_k^i(x) = a_k^i \log\left(b_k^i x + 1\right). \tag{2.1}$$

This form, which has been used elsewhere ([1], [11], [69], [70], [71]) is shown in the schematic in Figure 2.2. The quantities a_k^i and b_k^i are prosumer specific constants, while x is its load. The quantity a_k^i is chosen to convert the utility into monetary units, while b_k^i governs the utility function's rate of saturation. It must be emphasized that the analytical treatment throughout this research can handle any other utility function with the above characteristics.

Prosumers in this framework may be equipped with their own PV panels, thus capable of generating an amount of energy g_k^i so that its net power demand is the difference between x and generation, g_k^i as,

$$p_k^i = x - g_k^i. \tag{2.2}$$

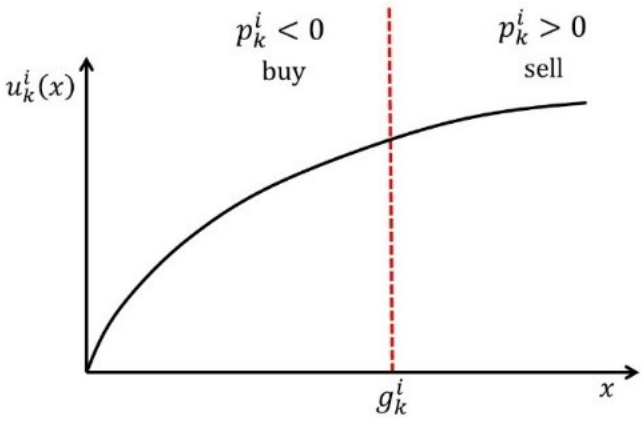

Figure 2.2: Utility of a prosumer as a function of energy consumed.

The sign of p_k^i indicates whether the prosumer receives or supplies energy to the grid. When $p_k^i > 0$, the prosumer's consumption is more than its generation ($x > g_k^i$); consequently, it must receive the additional energy from the grid. Conversely, when $p_k^i < 0$ the prosumer supplies the surplus energy to the latter. All parameters associated with each prosumer (a_k^i, b_k^i, g_k^i) as well as the choice of utility functions remain hidden from the rest of the grid.

As selfish agents, prosumers try to maximize their payoffs, which is the difference between the utility obtained from energy use and the cost of procurement. With c_k being the unit cost provided to the agent, its strategy can be formulated as the following constrained optimization problem,

Maximize w.r.t. p_k^i:

$$\pi_k^i(p_k^i) = u_k^i(p_k^i + g_k^i) - c_k p_k^i. \tag{2.3a}$$

Subject to:

$$p_k^i + g_k^i \geq 0. \tag{2.3b}$$

Differentiating the payoff π_k^i in Eqn. (2.3a) with respect to p_k^i, with u_k^i being as in Eqn. (2.1) and using the constraint in Eqn. (2.3b) and the load x in Eqn. (2.2), it can be shown that the demand is,

$$p_k^i = \max\left(\frac{a_k^i b_k^i - c_k}{c_k b_k^i}, 0\right) - g_k^i. \tag{2.4}$$

2.2.2 Aggregator

The aggregators in $\mathcal{A}$ are intermediary entities between their prosumers and the DSO, their role being primarily communicative. Each aggregator k receives a unit cost signal from the latter, that it sends to the agents in $\mathcal{G}_k$. The prosumers' response is the corresponding demand p_k^i as obtained from Eqn. (2.3), or with logarithmic utilities, from Eqn. (2.4). The social welfare at each aggregator is the sum of the utilities of all its prosumers, and given by,

$$\mathcal{W}_k(\mathbf{p}_k) = \sum_{i \in G_k} u_k^i(p_k^i + g_k^i)\,. \tag{2.5}$$

In the above expression, $\mathbf{p}_k = [p_k^i]_{i \in \mathcal{G}_k}$. Neglecting the constraint in Eqn. (2.3b) it can be seen that,

$$\frac{\partial}{\partial p_k}\mathcal{W}_k(\mathbf{p}_k) = c_k. \tag{2.6}$$

This shows that the aggregator k responds to the DSO's unit cost c_k with an aggregate energy demand $p_k = \mathbf{1}_{G_k}^{\mathrm{T}}\mathbf{p}_k$ ($G_k = |\mathcal{G}_k|$) such that the slope of $\mathcal{W}_k$ is c_k. This information from all aggregators allows the DSO to construct the gradient $\nabla_{\mathbf{p}}[\mathcal{W}_k(p_k)]_{k \in \mathcal{A}}$ required for its optimization algorithm (for further details, one is referred to [72]).

2.2.3 DSO

The DSO's role is in realizing the underlying optimization algorithm. It receives power demand as the vector $\mathbf{p} = [p_k]_{k \in \mathcal{A}}$ from the set $\mathcal{A}$ of aggregators and returns the DLMP $\mathbf{c} = [c_k]_{k \in \mathcal{A}}$ to the latter. Further details of the DSO are described in the next section.

2.3 Mathematical Model

2.3.1 Linearized AC Power Flow

The schematic in Figure 2.3 shows a segment of the radial distribution network. Each node is labeled with an index $k \in \mathcal{N}$. Line indices are identical to those of the nodes at their receiving ends. With p_k and q_k being the active and reactive power injected at any node k, the active and reactive power flowing through the line k are given by the following expressions,

$$P_k = p_k + \sum_{l \in \mathcal{D}(k)} L_l^P + \sum_{l \in \mathcal{D}(k) \cap \mathcal{A}} p_l\,, \tag{2.7a}$$

$$Q_k = q_k + \sum_{l \in \mathcal{D}(k)} L_l^Q + \sum_{l \in \mathcal{D}(k) \cap \mathcal{A}} q_l\,. \tag{2.7b}$$

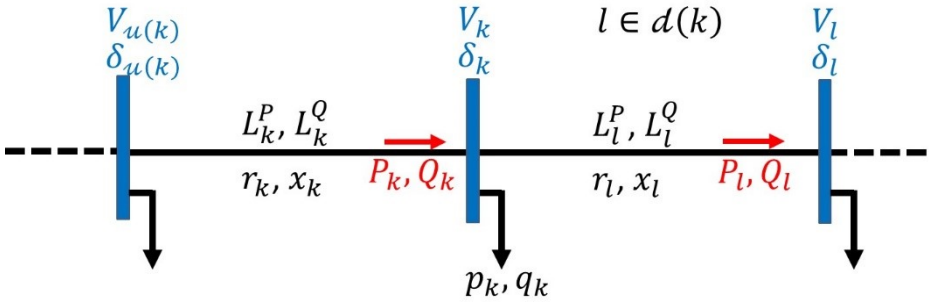

Figure 2.3: Distribution system radial branch model.

Here, $\mathcal{D}(k)$ is the set of all nodes that are downstream of node k. The quantities L_k^P and L_k^Q are the active and reactive line losses, that are computed as follows,

$$L_k^P = r_k \frac{P_k^2 + Q_k^2}{V_k^2}, \tag{2.8a}$$

$$L_k^Q = x_k \frac{P_k^2 + Q_k^2}{V_k^2}. \tag{2.8b}$$

In Eqn. (2.8), V_k is the voltage at node k while r_k and x_k are the corresponding line resistance and reactance. Letting $\boldsymbol{P} = [P_k]_{k \in \mathcal{A}}$ and $\boldsymbol{Q} = [Q_k]_{k \in \mathcal{A}}$, it follows from Eqn. (2.7) that,

$$\boldsymbol{P} = (\mathbf{I} + \mathbf{T})\mathbf{p} + \mathbf{T}\boldsymbol{L}^P, \tag{2.9a}$$

$$\boldsymbol{Q} = (\mathbf{I} + \mathbf{T})\mathbf{q} + \mathbf{T}\boldsymbol{L}^Q. \tag{2.9b}$$

In the above, the vector $\mathbf{p} = [p_k]_{k\in\mathcal{A}}$ may be regarded as an $|\mathcal{N}| \times 1$ vector with zeros occupying every place k where $k \notin \mathcal{A}$, $\mathbf{I}$ is the $N \times N$ identity matrix. The $N \times N$ tree matrix $\mathbf{T}$ therein is defined as,

$$[\mathbf{T}]_{k,l\in\mathcal{N}} = \begin{cases} 1, & l \in \mathcal{D}(k), \\ 0, & \text{otherwise.} \end{cases}$$

The real and reactive powers at the sending end of line k are given by the following expressions,

$$P_k + L_k^P = \frac{r_k V_{u(k)}\left(V_{u(k)} - V_k \cos\left(\delta_{u(k)} - \delta_k\right)\right)}{r_k^2 + x_k^2} + \frac{V_k V_{u(k)} \sin\left(\delta_{u(k)} - \delta_k\right)}{r_k^2 + x_k^2}, \tag{2.10a}$$

$$Q_k + L_k^Q = \frac{x_k V_{u(k)}\left(V_{u(k)} - V_k \cos\left(\delta_{u(k)} - \delta_k\right)\right)}{r_k^2 + x_k^2} + \frac{V_k V_{u(k)} \sin\left(\delta_{u(k)} - \delta_k\right)}{r_k^2 + x_k^2}. \tag{2.10b}$$

In Eqn. (10), δ_k is the voltage angle at node k and $u(k)$ is the index of the node that is immediately upstream of it.

The expressions in Eqn. (2.10) are linearized to simplify the grid constraints (see later). We adopt the linearized power flow model proposed in [3]. Assuming that $\left|V_{u(k)}\right|, |V_k| \approx 1\, p.u.$, $\delta_{u(k)} - \delta_k \approx 0$, the above equalities are approximated to yield,

$$P_k + L_k^P = b_k^r\left(V_{u(k)} - V_k\right) + b_k^x\left(\delta_{u(k)} - \delta_k\right), \tag{2.11a}$$

$$Q_k + L_k^Q = b_k^x\left(V_{u(k)} - V_k\right) - b_k^r\left(\delta_{u(k)} - \delta_k\right). \tag{2.11b}$$

Here,

$$b_k^r = \frac{r_k}{r_k^2 + x_k^2}, \quad b_k^x = \frac{x_k}{r_k^2 + x_k^2}.$$

The expressions in Eqn. (2.11) can be represented more concisely in the following manner. With $\boldsymbol{V}$ and $\boldsymbol{\delta}$ representing the $N \times 1$ vectors of all node voltages and angles, it can be shown that,

$$\begin{bmatrix} \boldsymbol{P} \\ \boldsymbol{Q} \end{bmatrix} = \mathbf{M}\begin{bmatrix} \boldsymbol{V} \\ \boldsymbol{\delta} \end{bmatrix} - \begin{bmatrix} \boldsymbol{L}^P \\ \boldsymbol{L}^Q \end{bmatrix} + \mathbf{N}. \tag{2.12}$$

In Eqn. (2.12),

$$\mathbf{M} = \begin{bmatrix} \mathbf{B}^r(\mathbf{D}^{\mathrm{T}} - \mathbf{I}) & \mathbf{B}^x(\mathbf{D}^{\mathrm{T}} - \mathbf{I}) \\ \mathbf{B}^x(\mathbf{D}^{\mathrm{T}} - \mathbf{I}) & -\mathbf{B}^r(\mathbf{D}^{\mathrm{T}} - \mathbf{I}) \end{bmatrix}, \tag{2.13}$$

$$\mathbf{N} = \begin{bmatrix} \mathbf{B}^r V_0 \mathbf{e} + \mathbf{B}^x \delta_0 \mathbf{e} \\ \mathbf{B}^x V_0 \mathbf{e} + \mathbf{B}^r \delta_0 \mathbf{e} \end{bmatrix}, \tag{2.14}$$

$$\mathbf{B}^r = \mathrm{diag}\left(\frac{r_k}{r_k^2 + x_k^2}\right), \tag{2.15a}$$

$$\mathbf{B}^x = \mathrm{diag}\left(\frac{x_k}{r_k^2 + x_k^2}\right). \tag{2.15b}$$

The $N \times N$ matrix $\mathbf{D}$, called the downstream matrix is given by,

$$[\mathbf{D}]_{k,l\in\mathcal{N}} = \begin{cases} 1, & l \in d(k), \\ 0, & \text{otherwise.} \end{cases}$$

The substation bus is indexed $0 \notin \mathcal{N}$. Its voltage V_0 and angle δ_0 are treated as constant quantities.

Node voltages in terms of the nodes' real and reactive power injections, can be obtained from Eqn. (2.9) and Eqn. (2.12).

$$\begin{bmatrix} \boldsymbol{V} \\ \boldsymbol{\delta} \end{bmatrix} = \mathbf{C}\left(\begin{bmatrix} \mathbf{T}+\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{T}+\mathbf{I} \end{bmatrix}\begin{bmatrix} \boldsymbol{L}^P \\ \boldsymbol{L}^Q \end{bmatrix} - \begin{bmatrix} \mathbf{B}^r V_0 \mathbf{e} + \mathbf{B}^x \delta_0 \mathbf{e} \\ \mathbf{B}^x V_0 \mathbf{e} + \mathbf{B}^r \delta_0 \mathbf{e} \end{bmatrix}\right) + \mathbf{C}\begin{bmatrix} (\mathbf{T}+\mathbf{I})\mathbf{A} & \mathbf{0} \\ \mathbf{0} & (\mathbf{T}+\mathbf{I})\mathbf{A} \end{bmatrix}\begin{bmatrix} \mathbf{p} \\ \mathbf{q} \end{bmatrix}. \tag{2.16}$$

In Eqn. (2.16), $\mathbf{e}$ is a vector with a 1 appearing as its first entry and all others being zeroes. The matrix $\mathbf{C}$ is obtained according to,

$$\mathbf{C} = \begin{bmatrix} \mathbf{B}^r(\mathbf{D}^{\mathrm{T}} - \mathbf{I}) & \mathbf{B}^x(\mathbf{D}^{\mathrm{T}} - \mathbf{I}) \\ \mathbf{B}^x(\mathbf{D}^{\mathrm{T}} - \mathbf{I}) & -\mathbf{B}^r(\mathbf{D}^{\mathrm{T}} - \mathbf{I}) \end{bmatrix}^{-1}. \tag{2.17}$$

By applying Taylor's series expansion around the reference points, $\mathbf{p}_0$, $\mathbf{q}_0$, $\boldsymbol{L}_0^P$, and $\boldsymbol{L}_0^Q$, linear expressions for the losses can be obtained from Eqn. (2.2) as shown below,

$$\boldsymbol{L}^P = \boldsymbol{L}_0^P + \mathbf{J}_P^{L\,\mathrm{T}}(\mathbf{p} - \mathbf{p}_0) + \mathbf{J}_P^{L\,\mathrm{T}}\mathrm{diag}(\boldsymbol{\theta})^{-1}(\mathbf{q} - \mathbf{q}_0), \tag{2.18a}$$

$$\boldsymbol{L}^Q = \boldsymbol{L}_0^Q + \mathbf{J}_Q^{L\,\mathrm{T}}(\mathbf{p} - \mathbf{p}_0) + \mathbf{J}_Q^{L\,\mathrm{T}}\mathrm{diag}(\boldsymbol{\theta})^{-1}(\mathbf{q} - \mathbf{q}_0). \tag{2.18b}$$

The matrices $\mathbf{J}_P^L$ and $\mathbf{J}_Q^L$ in (18) are Jacobians of the losses in (8),

$$\mathbf{J}_P^L = \left[\frac{\partial L_l^P}{\partial p_k}\right]_{l\in\mathcal{N},k\in\mathcal{A}}, \qquad \mathbf{J}_Q^L = \left[\frac{\partial L_l^Q}{\partial q_k}\right]_{l\in\mathcal{N},k\in\mathcal{A}}.$$

2.3.2 Jain's Fairness Index

The Jain's fairness index is defined as the squared inner product of the unit vectors $\hat{\mathbf{x}}$ and $\hat{\mathbf{1}}$, where $\hat{\mathbf{x}}$ is along the direction of $\mathbf{x}$ and $\hat{\mathbf{1}}$ has identical elements. The following is an expression for the Jain's index,

$$J(\mathbf{x}) = \langle \hat{\mathbf{x}}, \hat{\mathbf{1}} \rangle^2. \tag{2.19}$$

In Eqn. (2.19), $\mathbf{x} = [x_i]_{i=1:n}$ each x_i is an amount of resource allocated to any user i in a set of n users. It lies in the interval $[0,1]$ with values closer to unity indicating more fairness.

Jain's index can be applied to the present context in a variety of ways. The simplest manner to implement fairness would be to replace each x_i above, with an aggregator power p_k. In this manner, the fairest allocation would be when all aggregators receive an equal amount of power. Unfortunately, this over-simplistic version of fairness does not account for the difference in the numbers of prosumers in each aggregator. However, one important insight can be gained from this observation. Supplying resources to m out of the n users ($m < n$) and allocating $x_i = 0$ to the remaining $n - m$ would still lead to maximum of the index when the m users receive the same amount. Hence, disregarding the aggregators that supply energy does not affect optimal fairness among the remaining users.

In this research, Jain's fairness is determined as,

$$J(\mathbf{n} \circ \mathbf{p}) = \frac{1}{\|\mathbf{z}\|_1} \frac{(\mathbf{n}^{\mathrm{T}}\mathbf{p})^2}{\|\mathbf{n} \circ \mathbf{p}\|^2}. \tag{2.20}$$

In the above expression and everywhere else, the operator $\|\cdot\|$ unless subscripted, is the L_2 norm. The quantity $\mathbf{z}$ is a logical vector of 0s and 1s and is given by,

$$\mathbf{z} = [p_k > 0]_{k \in \mathcal{A}}. \tag{2.21}$$

Hence, $\|\mathbf{z}\|_1$ is the number of aggregators that receive energy from the grid. Those that supply energy ($p_k < 0$) are set aside from fairness considerations. Although all aggregators could readily be used, precluding sellers demonstrates the flexibility of the proposed framework. In addition, as suggested in [69], in aggregators that sell energy, prosumers may collectively place bids to maximize their payoffs, applying their own fairness criteria. Moreover, the analytical treatment of Jain's index has been restricted to the case when the argument in $J(\cdot)$ is confined to the non-negative orthant. Replacing $\mathbf{z}$ with $\mathbf{1}$ would allow aggregators that supply energy to also be included in determining fairness. In Eqn. (2.21), the vector $\mathbf{n}$ is obtained as the following Hadamard product,

$$\mathbf{n} = \mathbf{c}^{\circ -1} \circ [G_k]_{k \in \mathcal{A}}^{\circ -1} \circ \mathbf{z}\,. \tag{2.22}$$

In the absence of any information regarding the size or electricity needs of the household associated with each prosumer, it is assumed that all prosumers have identical demands. The quantity $G_k = |\mathcal{G}_k|$ is present in the denominator in order to allocate power to the aggregators that

are roughly in proportion to their numbers of prosumers. Devising $\mathbf{n}$ can be refined further to include other information, such as household sizes, monetary values of the homes, historical consumption averages, etc., none of which have been considered.

The formulation provided in Eqn. (2.21) and Eqn. (2.22) applies proportional fairness to the remaining aggregators. Proportional fairness, a game theoretic concept where agents in an aggregator choose to pay more for energy are allocated greater amounts of power. Proportional fairness, whose effectiveness has been shown in [73], is imposed because in Eqn. (2.22) the vector $\mathbf{n}$ contains the vector of unit costs $\mathbf{c}$ in its denominator. Proportional fairness is achieved through the use of $\mathbf{n} \circ \mathbf{p}$ as the argument to $J(\cdot)$, which normalizes the energy demands of the aggregators with respect to their unit costs.

2.3.3 DSO Mechanism

The objective of the DSO is two-fold. It primarily attempts to maximize the social welfare $\mathbf{1}_A^{\mathrm{T}}[\mathcal{W}_k(p_k)]_{k\in\mathcal{A}}$ of all prosumers present in the grid. Next, it tries to price the agents as fairly as possible. Hence, the DSO's objective consists of a social welfare term, and an optional regularization term which is the Jain's index in Eqn. (2.20), weighted appropriately. Accordingly, the DSO mechanism's objective can be formulated as the following constrained optimization problem,

Maximize w.r.t. $\mathbf{p}$,

$$\Omega(\mathbf{p}) = \mathbf{1}_A^{\mathrm{T}}[\mathcal{W}_k(p_k)]_{k\in\mathcal{A}} + \frac{C}{2}J(\mathbf{n}\circ\mathbf{p}). \tag{2.23a}$$

Subject to,

$$-\mathbf{C}^V\mathbf{p} + \mathbf{c}_l^V \leq \mathbf{0}, \tag{2.23b}$$

$$\mathbf{C}^V\mathbf{p} + \mathbf{c}_u^V \leq \mathbf{0}, \tag{2.23c}$$

$$\mathbf{C}^S\mathbf{p} + \mathbf{c}_0^S \leq \mathbf{0}, \tag{2.23d}$$

$$\mathbf{c}^{P_0\,\mathrm{T}}\mathbf{p} + c_0^{P_0} - P_0 = 0, \tag{2.23e}$$

$$-\mathbf{c}^{\mathrm{T}}\mathbf{p} + c_0 P_0 \leq 0. \tag{2.23f}$$

The linear equality and inequality constraints in Eqn. (23*b*) – Eqn. (23*f*) are obtained directly from the linearization described earlier in this section, with appropriate rearrangements that are shown

in Appendix-B. As this research used real power to determine unit costs, they have been expressed compactly in terms of the real power, which is the primal variable, $\mathbf{p} = [p_k]_{k\in\mathcal{A}}$. The inequalities in Eqn. (23*b*) and Eqn. (23*c*) restrict the node voltage deviations to lie within $V_0 \pm \epsilon$, where V_0 is the substation voltage. Likewise, real and reactive line flow limits are imposed by means of Eqn. (23*d*). The equality appearing in Eqn. (23*e*) is the power balance constraint that ensures that P_0, the total power supplied to the DSO by the wholesale market equals the sum of the total power demands of the grid's prosumers and the power losses. Lastly, Eqn. (23*f*) is the weak budget balance constraint that restricts the feasible region to one where the DSO's payment at a unit cost of c_0 does not exceed the total monetary amount that it receives from the aggregators.

With $\underline{\boldsymbol{\alpha}}, \overline{\boldsymbol{\alpha}}, \boldsymbol{\beta}, \lambda, \gamma$ being dual variables, and η (with appropriate subscripts), increment factors, the augmented Lagrange function corresponding to Eqn. (2.23) is given by,

$$\begin{aligned}
\mathfrak{L}_a(\mathbf{p}, \underline{\boldsymbol{\alpha}}, \overline{\boldsymbol{\alpha}}, \boldsymbol{\beta}, \lambda, \gamma) = \sum_{k\in\mathcal{A}} \mathcal{W}_k(p_k) + \frac{C}{2} J(\mathbf{n}\circ\mathbf{p}) \\
-\overline{\boldsymbol{\alpha}}^{\mathrm{T}}(\mathbf{C}^V\mathbf{p} + \mathbf{c}_u^V) - \frac{\eta_V}{2}(\mathbf{C}^V\mathbf{p} + \mathbf{c}_u^V)^{\mathrm{T}}(\mathbf{C}^V\mathbf{p} + \mathbf{c}_u^V)_+ \\
-\boldsymbol{\beta}^{\mathrm{T}}(\mathbf{C}^S\mathbf{p} + \mathbf{c}_0^S) - \frac{\eta_S}{2}(\mathbf{C}^S\mathbf{p} + \mathbf{c}_0^S)^{\mathrm{T}}(\mathbf{C}^S\mathbf{p} + \mathbf{c}_0^S)_+ \\
-\lambda\left(\boldsymbol{C}^{P_0{}^{\mathrm{T}}}\mathbf{p} + c_0^{P_0} - P_0\right) - \frac{\eta_P}{2}\left(\boldsymbol{C}^{P_0{}^{\mathrm{T}}}\mathbf{p} + c_0^{P_0} - P_0\right)^2 \\
-\gamma(-\mathbf{c}^{\mathrm{T}}\mathbf{p} + c_0P_0) - \frac{\eta}{2}(-\mathbf{c}^{\mathrm{T}}\mathbf{p} + c_0P_0)(-\mathbf{c}^{\mathrm{T}}\mathbf{p} + c_0P_0)_+. \qquad (2.24)
\end{aligned}$$

Equating its derivative $\nabla_{\mathbf{p}}\mathfrak{L}(\mathbf{p}, \boldsymbol{\lambda}, \boldsymbol{\xi}, \gamma)$ to zero, and using Eqn. (2.6), an incremental update rule for the unit cost is obtained,

$$\begin{aligned}
\mathbf{c}^{\mathrm{new}} \leftarrow -\mathbf{C}^{V^{\mathrm{T}}}\underline{\boldsymbol{\alpha}} - \eta_V\mathbf{C}^{V^{\mathrm{T}}}(-\mathbf{C}^V\mathbf{p} + \mathbf{c}_l^V)_+ + \mathbf{C}^{V^{\mathrm{T}}}\overline{\boldsymbol{\alpha}} + \eta_V\mathbf{C}^{V^{\mathrm{T}}}(\mathbf{C}^V\mathbf{p} + \mathbf{c}_u^V)_+ \\
+\mathbf{C}^{S^{\mathrm{T}}}\boldsymbol{\beta} + \eta_S\mathbf{C}^{S^{\mathrm{T}}}(\mathbf{C}^S\mathbf{p} + \mathbf{c}_0^S)_+ + \lambda\boldsymbol{C}^{P_0} + \eta_P\left(\boldsymbol{C}^{P_0}\mathbf{p}^{\mathrm{T}} + c_0^{P_0} - P_0\right)\boldsymbol{C}^{P_0{}^{\mathrm{T}}} \\
-\gamma\mathbf{c} - \eta\mathbf{c}(-\mathbf{c}^{\mathrm{T}}\mathbf{p} + c_0P_0)_+ - \frac{C}{2}\nabla_{\mathbf{p}}J(\mathbf{n}\circ\mathbf{p}). \qquad (2.25)
\end{aligned}$$

In the above expression, $\nabla_{\mathbf{p}}J$ is the derivative of Jain's index in (20), with its argument $\mathbf{n}\circ\mathbf{p}$ restricted to the non-negative orthant. It is given by the expression given below,

$$\nabla_{\mathbf{p}} J(\mathbf{n} \circ \mathbf{p}) = \frac{2}{\|\mathbf{n} \circ \mathbf{p}\|} \mathbf{n} \circ \left(\sqrt{J(\mathbf{n} \circ \mathbf{p})} \frac{\mathbf{1}}{\|\mathbf{z}\|_1} - J(\mathbf{n} \circ \mathbf{p}) \frac{\mathbf{n} \circ \mathbf{p}}{\|\mathbf{n} \circ \mathbf{p}\|} \right). \tag{2.26}$$

The unit cost $\mathbf{c}$ is initialized at the beginning of the optimization algorithm. Using Eqn. (2.25) this cost is updated. Upon termination, the DLMP components can be readily obtained as the terms in the update rule for $\mathbf{c}^{\text{new}}$ in Eqn. (2.25). These are the voltage component, $\mathbf{c}_V$, the congestion component, $\mathbf{c}_C$, as well as the energy and loss component $\mathbf{c}_{E+L}$. Additionally, regularization introduces a new fairness component, $\mathbf{c}_F$. These are as follows,

$$\mathbf{c}_V = -\mathbf{C}^{V\mathrm{T}} \underline{\boldsymbol{\alpha}} - \eta_V \mathbf{C}^{V\mathrm{T}} (-\mathbf{C}^V \mathbf{p} + \mathbf{c}_l^V)_+ + \mathbf{C}^{V\mathrm{T}} \overline{\boldsymbol{\alpha}} + \eta_V \mathbf{C}^{V\mathrm{T}} (\mathbf{C}^V \mathbf{p} + \mathbf{c}_u^V)_+, \tag{2.27a}$$

$$\mathbf{c}_C = \mathbf{C}^{S\mathrm{T}} \boldsymbol{\beta} + \eta_S \mathbf{C}^{S\mathrm{T}} (\mathbf{C}^S \mathbf{p} + \mathbf{c}_0^S)_+, \tag{2.27b}$$

$$\mathbf{c}_{E+L} = \lambda \boldsymbol{C}^{P_0} + \eta_P \left(\boldsymbol{C}^{P_0} \mathbf{p}^{\mathrm{T}} + c_0^{P_0} - P_0 \right) \boldsymbol{C}^{P_0\mathrm{T}}, \tag{2.27c}$$

$$\mathbf{c}_F = \frac{C}{2} \nabla_{\mathbf{p}} J(\mathbf{n} \circ \mathbf{p}). \tag{2.27d}$$

The unit cost, $\mathbf{c}^{\text{new}}$ is therefore the sum of its components,

$$\mathbf{c}^{\text{new}} = \mathbf{c}_V + \mathbf{c}_C + \mathbf{c}_{E+L} + \mathbf{c}_F. \tag{2.28}$$

The budget balance terms $(-\gamma \mathbf{c} - \eta \mathbf{c}(-\mathbf{c}^{\mathrm{T}} \mathbf{p} + c_0 P_0)_+)$ are not included in Eqn. (2.28) because, in the simulations described in the next section, as expected the constraint was inactive upon convergence, leaving behind a small monetary surplus with the DSO.

In each iteration of the optimization algorithm, the dual variables are updated using dual gradient descent as follows,

$$\underline{\boldsymbol{\alpha}} \leftarrow \left[\underline{\boldsymbol{\alpha}} + \eta_V (-\mathbf{C}^V \mathbf{p} + \mathbf{c}_l^V) \right]_+, \tag{2.29a}$$

$$\overline{\boldsymbol{\alpha}} \leftarrow [\overline{\boldsymbol{\alpha}} + \eta_V (\mathbf{C}^V \mathbf{p} + \mathbf{c}_u^V)]_+, \tag{2.29b}$$

$$\boldsymbol{\beta} \leftarrow [\boldsymbol{\beta} + \eta_S (\mathbf{C}^S \mathbf{p} + \mathbf{c}_0^S)]_+, \tag{2.29c}$$

$$\lambda \leftarrow \lambda + \eta_P \left(\boldsymbol{C}^{P_0} \mathbf{p} + c_0^{P_0} - P_0 \right), \tag{2.29d}$$

$$\gamma \leftarrow \gamma + \eta (-\mathbf{c}^{\mathrm{T}} \mathbf{p} + c_0 P_0). \tag{2.29e}$$

The increment factors η_V, η_S, η_P and η above are adjusted in each iteration. The algorithm terminates only when the updates to $\mathbf{p}$ are such that $\|\Delta \mathbf{p}\|_1 \ll 1$ for several consecutive iterations.

The following proposition argues that termination occurs when the global maximum of $\Omega(\mathbf{p})$ is reached.

2.4 Simulation Results

The model used here was implemented on a modified IEEE 37-bus system as shown in Figure 2.4. There were 17 nodes with aggregators (shaded circles in Figure 2.4). For clarity, the aggregators were indexed separately as $\mathrm{A1}-\mathrm{A17}$. Three separate scenarios were created for this study.

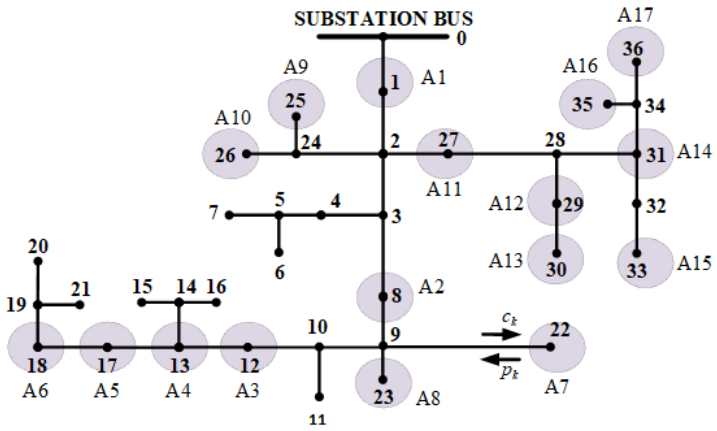

Figure 2.4: IEEE 37 bus system with aggregators indexed A1 – A17.

In Scenario-I all aggregators had $G_k = 10$ prosumers, without generation, but with a_k^i, b_k^i, generated randomly in each case. Scenario-II was similar to Scenario-I, except that the number of prosumers was doubled in aggregators $\mathrm{A3, A9, A11, A17}$ ($G_k = 20, k = 12,25,27,36$). In Scenario-III prosumers in aggregators $\mathrm{A8, A10, A14}$ $(k = 23, 26, 31)$ were equipped with PV generation. Their generations, g_k^i were obtained randomly. All simulations were performed using MATLAB. The increment factors were obtained in each iteration as $\eta = \|\mathbf{c}^{\mathrm{new}}\|^{-2}\eta_0$ with a separate η_0 for each factor η in the range 0.1 – 0.01. The mechanism was investigated for each scenario, both without and with fairness regularization ($C = 0, C = 0.4$ in $(23a)$).

Figure 2.5 (top) shows simulation results obtained from Scenarios-I, II, and III. The vertical bars are the power allocations of the aggregators that were obtained from the simulations. Those without fairness regularization appear in blue, (p_k) while those with regularization are in yellow

(p_k^*). The unit costs of the aggregators without fairness (c_k) and with fairness (c_k^*) are also provided in solid and dotted lines.

From Figure 2.4 it can be seen that aggregators $A1, A9, A10$ are positioned close to the substation bus. Consequently, when not regularized for fairness the mechanism outputs costs where these aggregators are sold energy at lower rates and therefore enjoy higher power allocations. In contrast, $A5, A6, A15, A16, A17$ experience higher unit costs and lower power allocations. Figure 2.5 shows how regularization helps in mitigating this adverse effect. Fairness causes aggregators to be charged in a more equitable manner.

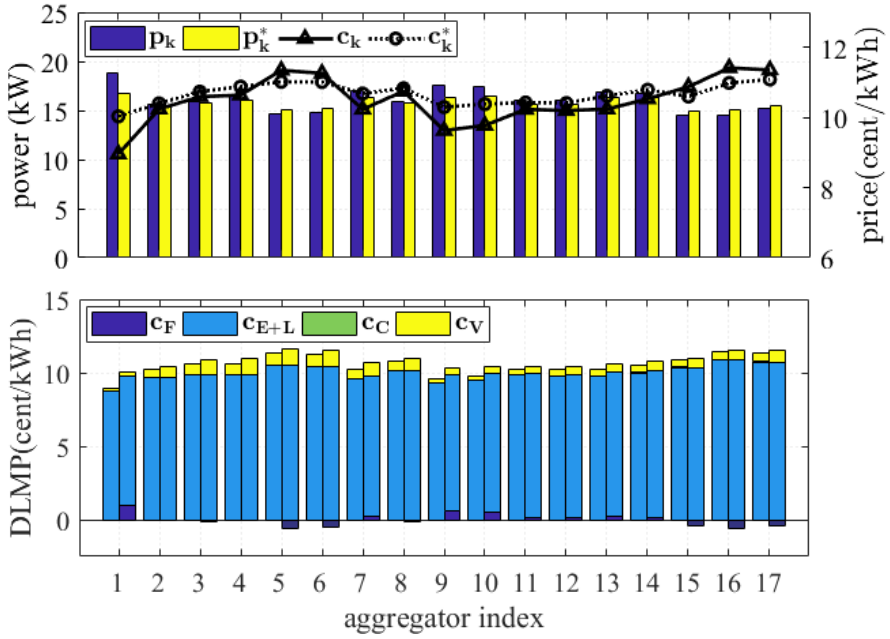

Figure 2.5: Results of Scenario-I showing unit costs and allocated power to each aggregator (top) and the DLMP components (bottom).

The breakdown of the unit costs into its DLMP components is also shown in Figure 2.5 (bottom). These are shown as stacked bars colored purple for the fairness component ($\mathbf{c}_F$), light blue for the unit cost due to energy usage and loss ($\mathbf{c}_{E+L}$), green for unit costs of congestion ($\mathbf{c}_C$), and yellow when the voltage limit constraints ($\mathbf{c}_V$) are active.

The DLMP components in Figure 2.5 sheds further insights into the differences in the unit costs. It is seen that the voltage components are very low for the three aggregators $A1, A9, A10$ located close to the substation. This is because voltage constraints are active further downstream in the grid, causing an increase in the voltage cost components in those aggregators

(A5, A6, A15, A16, A17). These are the aggregators that are furthest from the substation (Figure: 2.4). The mitigating effect of regularizing the DSO's objective is evident from the DLMP components in Figure 2.5. The algorithm provides cost discounts to the spatially disadvantaged aggregators, which is compensated by incrementing the unit costs of three aggregators in the substation's neighborhood.

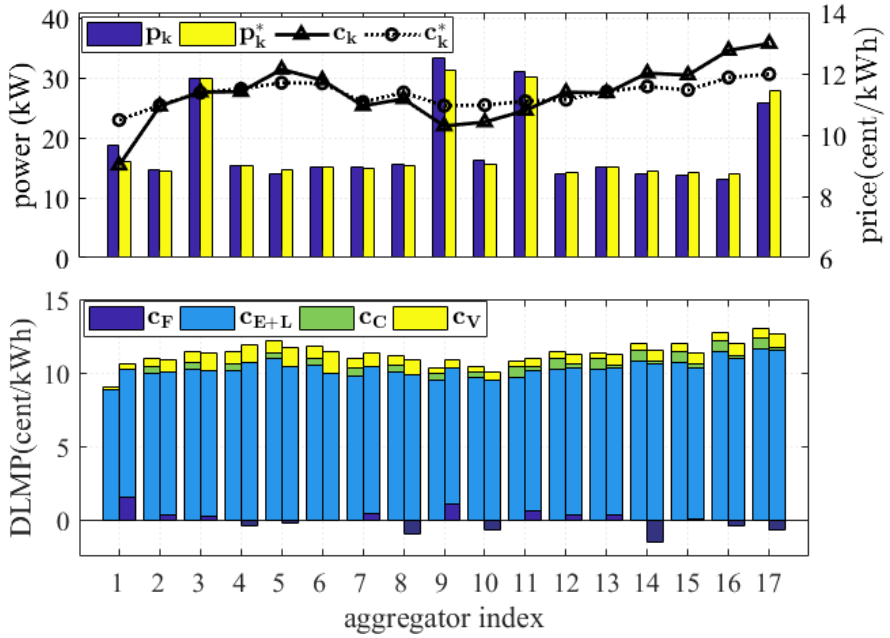

Figure 2.6: Results of Scenario-II showing unit costs and allocated power to each aggregator (top) and the DLMP components (bottom).

Figure 2.6 shows results of Scenario-II with twice the agents present in aggregators A3, A9, A11, A17 as elsewhere, allowing one to examine the effects of congestion. The rationale behind this choice of aggregators is due to the wide range of their locations vis-à-vis the substation, with A9, A11 being closest to it, A3 further away and A17 located at a large distance.

A similar pattern as before is observed in Figure 2.6, with distances having a severe impact on the unit costs. Increased loads in some aggregators cause congestion. Consequently, in the absence of regularization, aggregators yield higher unit costs for aggregators A4, A5, A6, A15, A16, A17 due to their distance than the others, as well as in comparison to what they were charged in Scenario-I.

The DLMP components elucidate the effect of higher congestion. Aggregators A4, A5, A6, A15, A16, A17 are priced at higher levels ($\mathbf{c}_C$). The significantly lower unit cost of A1

and to a lesser extent, A9, A10 due to their closeness to the substation, is evident. Supplementing the DSO's objective with Jain's index helps alleviate the pricing disparity. The previously advantaged aggregators see the highest unit costs due to the fairness component ($\mathbf{c}_F$). Conversely, those furthest away are able to increase their demands due to the lowered costs.

The effects of the penetration of PV generation in aggregators A8, A10, A14 diminishes the undesirable influence of congestion in voltage deviations throughout the distribution grid. Due to their PV generations, and resultant lower demands, the fairness component rewards them with highest drop in unit costs. This is seen in Figure: 2.7. Other aggregators also benefit from the introduction of PV generation in the grid in comparison the unit costs without fairness shown in Fig. 7. The congestion costs ($\mathbf{c}_C$) are uniformly lower than in the previous case. The DLMP components of the unit costs in this figure shows how regularization tries to rectify locational discrepancies.

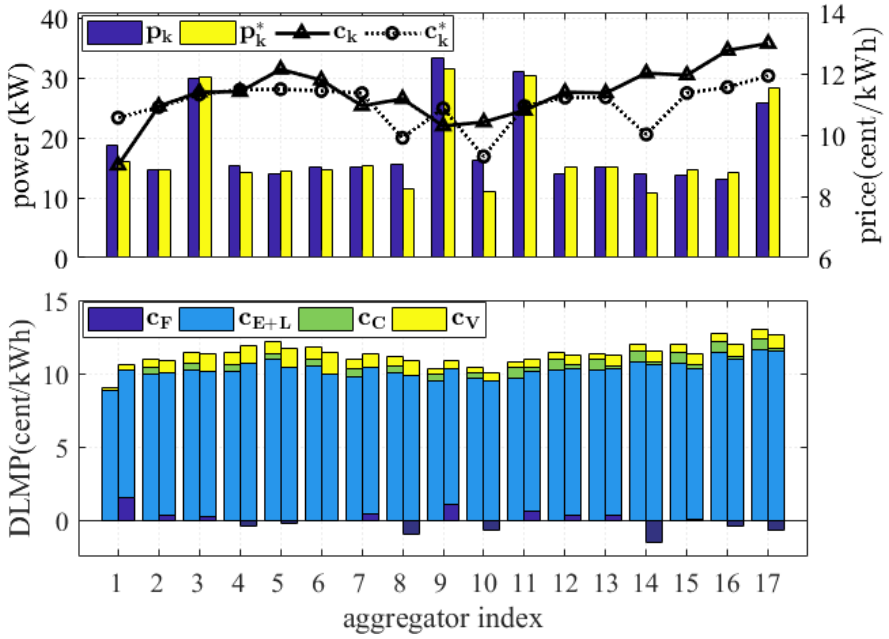

Figure 2.7: Results of Scenario-III showing unit costs and allocated power to each aggregator (top) and the DLMP components (bottom).

Lastly, the tradeoff between welfare and fairness, which is very well quantified in econometric and game-theory literature, is briefly addressed. Scenario-II was simulated when the regularization weight C was varied between $C = 0$ and $C = 0.5$ in increments of 0.02. The results are shown in Figure 2.8 clearly illustrating the tradeoff. Increasing fairness $J(\mathbf{n} \circ \mathbf{p})$ is associated

with a simultaneous decrease in the social welfare, $\mathbf{1}_A^{\mathrm{T}}[\mathcal{W}_k(p_k)]_{k\in\mathcal{A}}$, as seen in Fig. 8 (left). Figure 2.8 (right) shows the same phenomenon in terms of the price of fairness – the fraction reduction in efficiency. The term efficiency here refers to the ratio of the social welfare with regularization to that of its maximum attainable value sans regularization. The price of fairness is given by,

$$1 - \frac{J(\mathbf{n} \circ \mathbf{p})}{\max\limits_{\mathbf{p}} \mathbf{1}_A^{\mathrm{T}}[\mathcal{W}_k(p_k)]_{k\in\mathcal{A}}}.$$

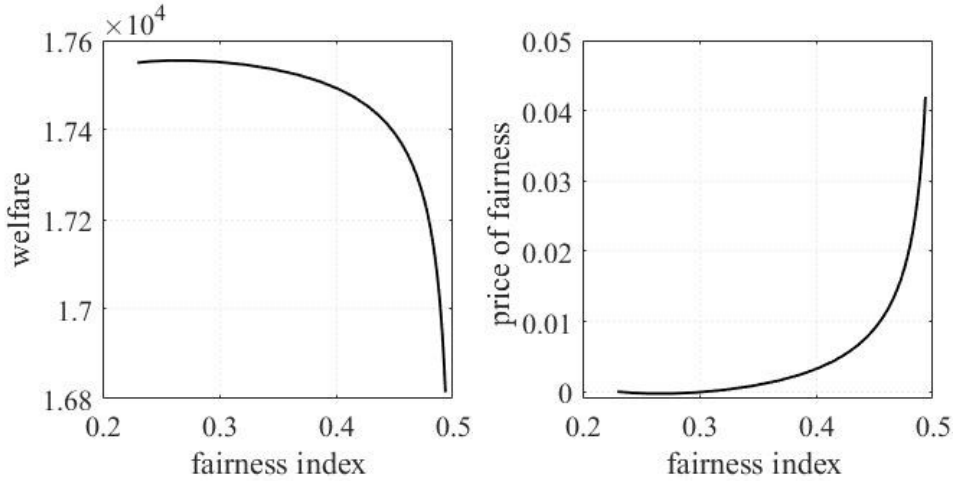

Figure 2.8: Total welfare and price of fairness vs. Jain's index

The presence of a 'knee' region in the trade-off curves in Figure 2.8 can be observed. Values of C beyond this region cause a sharp drop in welfare, but with relatively smaller increments in fairness. Fortunately, even in the extreme case ($C = 0.5$), the price of fairness is approximately equal to 0.04, which translates to a somewhat tolerable 4% reduction in the outcome's efficiency.

2.5 Conclusion

This research entails innovations along several directions. It is privacy preserving, in that it attains the global maximum of $\Omega(\mathbf{p})$ in Eqn. (23), without any knowledge about the prosumers' parameters. This is because the bidding process taking place between the prosumers and the aggregators allows the gradient $\nabla_{\mathbf{p}}\Omega(\mathbf{p})$ to be computed. The use of dual decomposition helps in dissociating the various DLMP components of the unit costs.

The application of Jain's fairness index in a distribution system and derivation of its component for DLMP is, to the best of the authors' knowledge, novel. It was shown to apportion energy to the aggregators in a more equitable manner. In addition, again to the best of the authors' knowledge, the use of Jain's index within the gradient descent algorithm is novel.

There are a few limitations of the research described here. The algorithm's rate of convergence was not investigated properly. When the initialization was entirely arbitrarily, the gradient descent would start at an infeasible region that was very far away from the active feasible manifold. However, the algorithm consistently reached the optimum in $\sim 10^2$ steps which is acceptable.

In extending the proposed approach to multiple (say, $T > 1$) time slots, each agent would receive a $T \times 1$ vector of costs $\mathbf{c}_k$ from the aggregator, and similarly respond with a vector of demands $\mathbf{p}_k^i$. Since this extended approach would require computing optimal schedules of energy storage elements and shiftable loads (e.g. PHEV, washer/dryer unit, etc.), the agent's scalar constrained optimization in Eqn. (3) whose optimum can be obtained in a straightforward manner as shown in Eqn. (4), would have to be replaced with a mixed integer convex programming algorithm. Although the DSO problem would still be formulated in the manner shown in Eqn. (23), the dimensionality of the primal variable $\mathbf{p}$ would increase T-fold. The resulting increase in computational overheads needs to be investigated. The authors believe that using the primal and dual variables of earlier time slots instead of arbitrarily initializing the algorithm each time, would achieve several fold speedups. Jain's index would also be required to quantify fairness over T time slots. One plausible approach would be to treat the composite Jain's index as one that is divided into T non-overlapping time partitions. The partition-irrelevancy of Jain's index can then be readily invoked. Jain's index has been studied extensively for fairness in allocating multiple resources (cf. [75]). Hence, another approach would be to treat energy during each time slot as a distinct resource. Further studies would reveal the most suitable method.

Linearization was used for mathematical convenience, computational simplicity, and as linear constraints guarantee unique maxima. The approximation error in the output must be quantified for a more thorough assessment of the approach's performance, and theoretical upper bounds established. However, it may be noted that the outputs of some simulations were compared with actual power flow. The largest errors were in the line voltages, which was acceptable at the order of 10^{-3}.

A more rigorous investigation into Jain's index needs to be carried out. In particular, the choice of the parameter C in Eqn. (23) has not been examined here; the range used in the simulations being rather arbitrary. A value of C must be picked with utmost care to avoid the 'knee' threshold (Figure 2.8), beyond which efficiency drops steeply. The results discussed in the previous section are *ad hoc*, offering no insights into what appropriate choices of C might look like in other energy grids. The sole recommendation here is that it be scaled in proportion to the dimensionality of the argument to Jain's index. Fortunately, the theoretical treatment of the tradeoff between welfare and fairness in [16] may offer invaluable insights for our purpose. It derives an expression for the maximum weight that can be assigned to fairness vis-à-vis welfare, beyond which the Pareto-optimality of the efficiency-fairness tradeoff can no longer be guaranteed. In other words, for an energy demand $\mathbf{p}$ that our approach yields with a large C, there might exist another demand $\mathbf{p}'$ that is both fairer as well as more efficient than $\mathbf{p}$! It remains unclear whether the cessation of Pareto-optimality in [16] bears any relationship to the 'knee' here.

Chapter 3 - Pareto-Optimal Energy Allocation Mechanism in Transactive Energy Market

In a grid-constrained transactive distribution system market, distribution locational marginal pricing (DLMP) is influenced by the distance from the substation to an energy user, thereby causing households that are further away from the substation to be charged more. The Jain's index of fairness, which has been recently applied to alleviate this undesirable effect in efficient energy allocations, is used in this research to quantify fairness. It is shown that the Jain's index is strictly quasi-concave. A bilevel distributed mechanism is proposed, where at the lower level, auction mechanisms are invoked simultaneously at each aggregator to obtain energy costs under market equilibrium conditions. A constrained multi-gradient ascent algorithm, Augmented Lagrangian Multigradient Approach (ALMA), is proposed for implementation at the upper level to attain energy allocations that represent tradeoffs between efficiency and fairness. Theoretical issues pertaining to ALMA as a generic algorithm for constrained vector optimization are considered. It is shown that when the objectives are restricted to be strictly quasi-concave functions and if the feasible region is convex, ALMA converges towards global Pareto optimality. The overall effectiveness of the proposed approach is confirmed through a set of MATLAB simulations implemented on a modified IEEE 37-bus system platform.

3.1 Introduction

In this research, a vector optimization algorithm is proposed to simultaneously maximize efficiency and fairness, the latter being quantified in terms of the Jain's index of fairness. The approach is an extension of the gradient ascent algorithm used in scalar optimization. The algorithm is capable of handling physical and other constraints imposed by the grid. Vector optimization is applied at the upper level of a bilevel framework and implemented by the DSO. The lower level incorporates distributed auction algorithms where energy users participate as bidding agents, is implemented by the aggregators.

A list of abbreviations for this chapter is provided in Appendix A.

3.2 Framework

Figure 3.1 is a schematic of the bilevel framework. The upper level mechanism is implemented by the DSO, which possesses physical information pertaining to the distribution grid. It communicates power allocations p_k from each aggregator, $k \in \mathcal{A}$, and receives equilibrium unit costs c_k from them. Only a subset $\mathcal{A}$ of $\mathcal{N}$ are aggregators. Each aggregator k contains a set $\mathcal{G}_k$ of prosumers within a physical neighborhood. The information flow between an aggregator and its agents $i \in \mathcal{G}_k$ are energy allocations, p_k^i as well as unit costs, c_k^i.

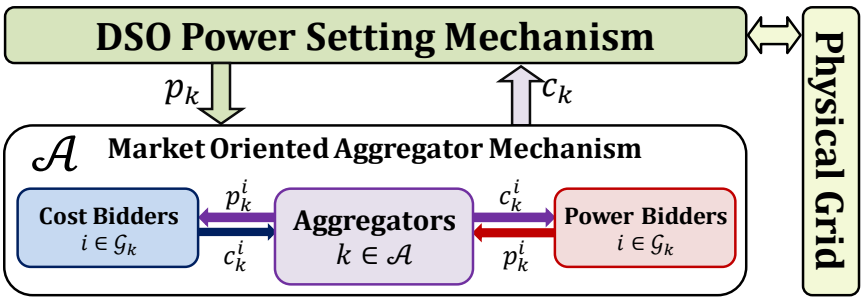

Figure 3.1: Schematic of market-driven bilevel mechanism

3.2.1 Aggregator Mechanism

There are two sets of agents in each aggregator k – the set $\mathcal{G}_k^{\mathrm{p}}$ of power bidders that receive unit costs and return power bids, as well as the set of cost bidders $\mathcal{G}_k^{\mathrm{c}}$ that receive power allocations and return cost bids. Both sets contain selfish agents that place bids to maximize their own payoffs, $u_k^i(p_k^i + g_k^i) - c_k^i p_k^i$. As shown in Figure 3.2, the utility functions $u_k^i(\cdot)$ are assumed to be strictly quasiconcave, monotonically increasing, differentiable, and includes the origin (as in [76]). The quantity g_k^i is the PV generation. A similar assumption of quasiconcave utilities has been adopted in [11] for energy trade between vehicles and the grid.

The auction algorithm is outlined below. The quantity S_k is called the supply and is the sum of the power p_k supplied by the DSO and those that the agents in $\mathcal{G}_k^{\mathrm{p}}$ are willing to sell to the aggregator at unit cost c_k. Similarly, the quantity R_k is the total monetary revenue that the aggregator will garner from the buying agents in $\mathcal{G}_k^{\mathrm{c}}$. Although these can be initialized in various possible ways, e.g. randomly, the number of iterations can be reduced drastically if the converged

values from a previous auction (of aggregator k) are used. The step where they as well as p_k^i, c_k^i, A_k and R_k are initialized is excluded from the outline of the auction.

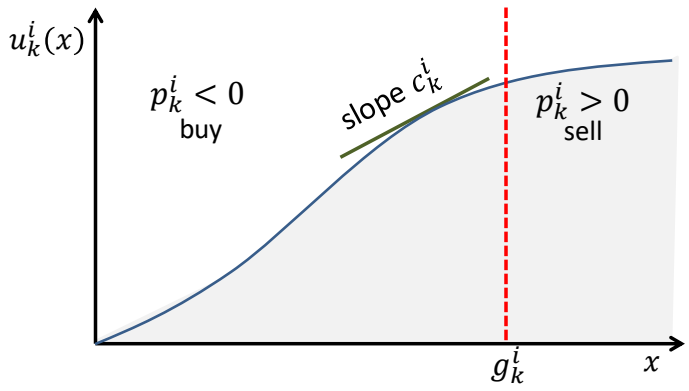

Figure 3.2: Typical quasiconcave utility function of agent $i \in \mathcal{G}_k$

Algorithm 3.1: Aggregator Auction

receive p_k from DSO

until (market equilibrium) **do**

1. $\mathcal{G}_k^{\mathrm{p}} \leftarrow \mathcal{G}_k^{\mathrm{p}} \backslash \{i \in \mathcal{G}_k^{\mathrm{p}} | p_k^i \geq 0\}$
   $\mathcal{G}_k^{\mathrm{C}} \leftarrow \mathcal{G}_k^{\mathrm{C}} \cup \{i \in \mathcal{G}_k^{\mathrm{p}} | p_k^i \geq 0\}$
2. $\mathcal{G}_k^{\mathrm{C}} \leftarrow \mathcal{G}_k^{\mathrm{C}} \backslash \{i \in \mathcal{G}_k^{\mathrm{p}} | c_k^i < c_k\}$
   $\mathcal{G}_k^{\mathrm{p}} \leftarrow \mathcal{G}_k^{\mathrm{p}} \cup \{i \in \mathcal{G}_k^{\mathrm{p}} | c_k^i < c_k\}$
3. $c_k \leftarrow \frac{R_k}{A_k}$
4. $\forall i \in \mathcal{G}_k^{\mathrm{P}}$: $p_k^i \leftarrow \underset{x}{\mathrm{argmax}}\left(u_k^i(x + g_k^i) - c_k x\right)$
5. $S_k \leftarrow p_k - \mathbf{1}_{|\mathcal{G}_k^{\mathrm{P}}|}^{\mathrm{T}} [p_k^i]_{i \in \mathcal{G}_k^{\mathrm{p}}}$
6. $\forall i \in \mathcal{G}_k^{\mathrm{C}}$: $p_k^i \leftarrow \frac{c_k^i p_k^i}{R_k} A_k$
7. $\forall i \in \mathcal{G}_k^{\mathrm{C}}$: $c_k^i \leftarrow \underset{x}{\mathrm{argmax}}\left(u_k^i(x + g_k^i) - x p_k^i\right)$
8. $R_k \leftarrow \mathbf{1}_{|\mathcal{G}_k^{\mathrm{C}}|}^{\mathrm{T}} [p_k^i]_{i \in \mathcal{G}_k^{\mathrm{C}}} \circ [c_k^i]_{i \in \mathcal{G}_k^{\mathrm{C}}}$

end do

send c_k to DSO

At the beginning of each iteration, the aggregator reassigns to $\mathcal{G}_k^{\mathrm{C}}$ any agent i that was previously in $\mathcal{G}_k^{\mathrm{P}}$ but intends to buy power $p_k^i \geq 0$ (step 1). Likewise, it transfers any agent $i \in \mathcal{G}_k^{\mathrm{P}}$ that has placed a unit cost bid $c_k^i < c_k$ to $\mathcal{G}_k^{\mathrm{C}}$ (step 2). In step 3, the aggregator level unit cost c_k is computed anew as the ratio of revenue R_k, to supply S_k. Next, the aggregator holds an auction within all sellers in $\mathcal{G}_k^{\mathrm{p}}$ (step 4) and receives as bids, the amounts of power p_k^i that they are willing to sell at the uniform rate, c_k. Following the sellers' auction, it updates the value of A_k. In step 6, which is referred to as *proportionally fair allocation* [1], [77], the aggregator divides the total power supply available S_k among the agents in $\mathcal{G}_k^{\mathrm{C}}$ in proportion to the total monetary amount that they are willing to pay. It holds an auction with the agents in $\mathcal{G}_k^{\mathrm{C}}$ bidding new values of c_k^i (step 7) In step 8, the aggregator updates R_k using the received bids, by summing the products $c_k^i p_k^i$ over all agents in $\mathcal{G}_k^{\mathrm{C}}$.

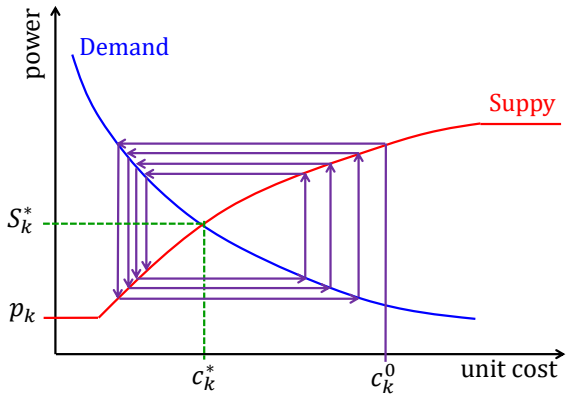

Figure 3.3: Convergence towards fixed point of aggregator auction

As seen in steps 1 and 2 of each auction iteration, the agents are assigned as buyers or sellers using their bids from the preceding iteration. This is different from earlier versions of the algorithms in [71], [34]. Assuming that no agent is reassigned in steps 1 and 2, the aggregator mechanism's convergence towards a fixed point is shown in Figure 3.3 It shows the supply S_k of the sellers in response to a cost c_k (red curve). The demand (blue curve) is the ratio $\frac{R_k}{S_k}$ after the buyers have placed their bids. Starting from an initial cost of $c_k = c_k^0$, the auction converges to $c_k = c_k^*$ in the counter-clockwise direction. It may be observed that a steeper supply curve would render the fixed point unstable. Divergence can be easily detected within two iterations of the

auction, in which case the mechanism can be implemented in the clockwise direction for convergence. As this situation is unlikely to happen in any realistic setting (and in our simulations), it has not been elaborated further.

The aggregator's total utility is given by,

$$\mathcal{U}_k(p_k) = \mathbf{1}_{G_k}^{\mathrm{T}}\left[u_k^i(p_k^i + g_k^i)\right]_{i\in\mathcal{G}_k} \tag{3.1}$$

Theorem 1 below shows that the aggregators reach an equilibrium unit cost $\mathbf{c}$ that is equal to $\nabla_{\mathbf{p}}\mathcal{W}(\mathbf{p})$, where $\mathcal{W}(\mathbf{p}) = \mathbf{1}_A^{\mathrm{T}}[\mathcal{U}_k(p_k)]_{k\in\mathcal{A}}$ is the welfare. In [76] this gradient is referred to as the marginal benefit. It will be assumed that the agents do not bid strategically so that the unit cost c_k is independent of the placed bid $x = \{p_k^i, c_k^i\}$, i.e. $\frac{\partial c_k}{\partial x} = 0$. The proof of Theorem 1 is more straightforward than and distinct from the indirect one in [78], where the statement of Theorem 1 was shown to be a limiting case of virtual bidding.

Theorem-1. *At the fixed point of the auction in aggregator k, the equilibrium cost is such that,*

$$\nabla_{\mathbf{p}}\mathcal{W}(\mathbf{p}) = \mathbf{c}. \tag{3.2}$$

Proof: Consider a sellers' bidding strategy as shown in step 4. If the bid is x its payoff is $u_k^i(x + g_k^i) - c_k x$. The payoff is maximum when its derivative with respect to x is zero; so it places a power bid such that $\frac{\partial}{\partial x}\left[u_k^i(x + g_k^i) - c_k x\right]_{x=p_k^i} = 0$, whence $u_k^{i\,\prime}(p_k^i + g_k^i) = c_k$.

Next, consider a buyer's bidding at any intermediate iteration. If the buyer responds to an allocation p_k^i with a cost bid of x, from proportional allocation, its share of the total power will be $\frac{S_k p_k^i}{R_k}$ in the next iteration. Assuming a large G_k such that S_k and R_k can be treated as constant with respect to the agent's bidding strategy, the bid is placed to maximize the overall payoff, $u_k^i\left(\frac{S_k p_k^i}{R_k}x + g_k^i\right) - x p_k^i$. This takes place in step 7. The derivative with respect to cost is $\frac{\partial}{\partial x}\left[u_k^i\left(\frac{S_k p_k^i}{R_k}x + g_k^i\right) - x p_k^i\right]_{x=c_k^i}$. Equated it to zero, we get, $\frac{S_k p_k^i}{R_k}u_k^{i\,\prime}\left(\frac{S_k p_k^i}{R_k}c_k^i + g_k^i\right) = p_k^i$. At the fixed point, in step 6 we must have, $p_k^i = \frac{c_k p_k^i}{R_k}S_k$, so that $c_k^i = \frac{R_k}{S_k}$. Under these circumstances the sellers bid is such that $u_k^{i\,\prime}(p_k^i + g_k^i) = c_k$.

Therefore it is seen that the equality $u_k^{i\,\prime}\left(p_k^i + g_k^i\right) = c_k$ applies to buyers and sellers. The statement of the theorem follows directly since $p_k = \mathbf{1}_{G_k}^{\mathrm{T}}[p_k^i]_{i\in\mathcal{G}_k}$.

■

3.2.2 Constraints

Let P_0 be the total energy that the DSO receives from external sources at a unit cost c_0. The constraints imposed on ALMA are as follows. The voltages at all nodes in $\mathcal{N}$ must remain within their minimum and maximum limits (lower/upper voltage deviation constraints). The active and reactive power flows in the lines must not exceed their capacities (capacity limit constraint). Additionally, P_0 must equal the sum of the energy delivered to the aggregators and the losses occurring at the lines (energy balance condition). Lastly, the amount that the DSO must pay to external sellers must not exceed the total revenue obtained from the aggregator (budget balance condition). With appropriate values of all coefficients, these constraints can be expressed concisely as follows (detailed derivation can be found in [79]),

$$\begin{cases} -\mathbf{C}^V\mathbf{p} + \mathbf{c}_l^V \le \mathbf{0}, & \text{(voltage deviation)} \\ \mathbf{C}^V\mathbf{p} + \mathbf{c}_u^V \le \mathbf{0}, & \text{(voltage deviation)} \\ \mathbf{C}^S\mathbf{p} + \mathbf{c}_0^S \le \mathbf{0}, & \text{(capacity limits)} \\ \mathbf{c}^{P_0\,\mathrm{T}}\mathbf{p} + c_0^{P_0} - P_0 = 0, & \text{(energy balance)} \\ -\mathbf{c}^{\mathrm{T}}\mathbf{p} + c_0 P_0 \le 0, & \text{(budget balance)} \end{cases} \tag{3.3}$$

Thus, the feasible set of allocations $\mathbf{p}$ is given by,

$$\mathcal{F}_{\mathbf{p}} \triangleq \left\{ \mathbf{p} \left| \begin{array}{c} -\mathbf{C}^V\mathbf{p} + \mathbf{c}_l^V \le \mathbf{0}, \mathbf{C}^V\mathbf{p} + \mathbf{c}_u^V \le \mathbf{0}, \\ \mathbf{C}^S\mathbf{p} + \mathbf{c}_0^S \le \mathbf{0}, \\ \mathbf{c}^{P_0\,\mathrm{T}}\mathbf{p} + c_0^{P_0} - P_0 = 0, -\mathbf{c}^{\mathrm{T}}\mathbf{p} + c_0 P_0 \le 0 \end{array} \right. \right\}. \tag{3.4}$$

3.2.3 Jain's Index of Fairness

The generic expression for Jain's index of fairness with argument $\mathbf{x}$ is as follows,

$$J(\mathbf{x}) = \frac{1}{\|\mathbf{1}\|^2}\frac{(\mathbf{1}^{\mathrm{T}}\mathbf{x})^2}{\mathbf{x}^{\mathrm{T}}\mathbf{x}}. \tag{3.5}$$

The main motivation behind the choice of Jain's index as the measure of fairness is its *Schur concavity*, which is expressed as $\mathbf{x} \succcurlyeq \mathbf{y} \Rightarrow J(\mathbf{x}) \geq J(\mathbf{y})$. In other words, if $\mathbf{x}$ *majorizes* $\mathbf{y}$ ($\mathbf{x} \succcurlyeq \mathbf{y}$), then $\mathbf{x}$ has a fairness index that is at least as high as that of $\mathbf{y}$. Majorization is explained as follows. Given the n dimensional vector $\mathbf{x}$, let $\mathbf{x}_d^\dagger$ denote the $d < n$ dimensional vector formed by taking the numerically smallest d elements of $\mathbf{x}$. For instance if $\mathbf{x} = [5\ 1\ 2\ 4\ 3]^{\mathrm{T}}$ then $\mathbf{x}_3^\dagger = [1\ 2\ 3]^{\mathrm{T}}$. The vector $\mathbf{y}_d^\dagger$ is obtained from $\mathbf{y}$ in an identical manner. We say that $\mathbf{x}$ majorizes $\mathbf{y}$ if and only if $\mathbf{1}_n^{\mathrm{T}}\mathbf{x} = \mathbf{1}_n^{\mathrm{T}}\mathbf{y}$ and $\mathbf{1}_d^{\mathrm{T}}\mathbf{x}_d^\dagger \geq \mathbf{1}_d^{\mathrm{T}}\mathbf{y}_d^\dagger,\ \forall d$. There is a more intuitive interpretation of this relationship, denoted as $\mathbf{x} \succcurlyeq \mathbf{y}$. Consider a pair of resource demand vectors, $\mathbf{x}$ and $\mathbf{y}$, with equal sums ($\mathbf{1}_n^{\mathrm{T}}\mathbf{x} = \mathbf{1}_n^{\mathrm{T}}\mathbf{y}$) allocated to n aggregators. The quantities $\mathbf{1}_d^{\mathrm{T}}\mathbf{x}_d^\dagger$ and $\mathbf{1}_d^{\mathrm{T}}\mathbf{y}_d^\dagger$ are the sums of the resources received by the d aggregators that have the least amount of resource allocated. Hence, $\mathbf{1}_d^{\mathrm{T}}\mathbf{x}_d^\dagger \geq \mathbf{1}_d^{\mathrm{T}}\mathbf{y}_d^\dagger$ implies that the resource-deprived aggregators collectively receive more resource through demand $\mathbf{x}$ than through demand $\mathbf{y}$. Hence, intuitively $\mathbf{x} \succcurlyeq \mathbf{y}$ means that $\mathbf{x}$ is fairer than $\mathbf{y}$.

In this research, the $G_k \times 1$ vector argument $\mathbf{x}$ of $J(\cdot)$ is determined as follows,

$$\mathbf{x} = \left[\frac{p_k}{c_k G_k}\right]_{k\in\mathcal{A}}. \tag{3.6}$$

Thus, each element consists of the energy p_k, normalized by the number of agents G_k and the unit costs c_k. Dividing the power p_k by G_k ensures that each aggregator receives energy in proportion to the total number of household agents in it. The presence of c_k in the denominator, is the DSO level version of proportional fairness [73], i.e. each aggregator allocated energy should be in proportion to the unit cost that the agents in it are willing to pay.

3.2.4 Constrained Vector Optimization Problem Formulation

The welfare $\mathcal{W}(\mathbf{p})$ of the DSO is the sum of the utilities of all aggregators in $\mathcal{A}$. The fairness, expressed as a function of $\mathbf{p} = [p_k]_{k\in\mathcal{A}}$ is denoted as $\mathcal{R}(\mathbf{p})$, which is equal to $J(\mathbf{x})$ as shown in Eqn. (3.5) where $\mathbf{x}$ is obtained from Eqn. (3.6). The vector objective $\boldsymbol{\Omega}(\mathbf{p})$ that ALMA must simultaneously maximize with respect to $\mathbf{p} \in \mathcal{F}_\mathbf{p}$ where $\mathcal{F}_\mathbf{p}$ is as in (4), is given by,

$$\boldsymbol{\Omega}(\mathbf{p}) \triangleq \begin{bmatrix}\mathcal{W}(\mathbf{p})\\ \mathcal{R}(\mathbf{p})\end{bmatrix}. \tag{3.7}$$

Theoretical issues related to CVOPs as well as details of ALMA are discussed next.

3.3 Proposed Approach

3.3.1 Theoretical Background

As mentioned earlier, ALMA although developed for this application, is a general-purpose algorithm for constrained vector optimization. Accordingly, as well as for conciseness, ALMA is discussed using more generic notation in Sections IV.A and IV.B. Without loss of generality, it is assumed that all objectives in the CVOP are to be maximized. Accordingly, let $\mathbf{f}: \mathbb{R}^n \to \mathbb{R}^m$ be the vector function to be maximized with respect to $\mathbf{x} \in \mathcal{F}_\mathbf{x}$, where $\mathcal{F}_\mathbf{x}$ is the feasible set,

$$\mathcal{F}_\mathbf{x} = \{\mathbf{x} | \mathbf{A}^\mathrm{T}\mathbf{x} + \mathbf{a} \leq \mathbf{0}, \mathbf{B}^\mathrm{T}\mathbf{x} + \mathbf{b} = \mathbf{0}\}. \tag{3.8}$$

Here, $\mathbf{A} \in \mathbb{R}^{n \times p}$, $\mathbf{a} \in \mathbb{R}^{p \times 1}$, $\mathbf{B} \in \mathbb{R}^{n \times q}$, $\mathbf{b} \in \mathbb{R}^{q \times 1}$. The feasible region $\mathcal{F}_\mathbf{p}$ introduced earlier in Eqn. (3.4) clearly fits the generic form in Eqn. (3.8). Scalar objectives in $\mathbf{f}(\cdot)$ are denoted as $f_j(\cdot), j \in \{1,2,\ldots,m\}$. It is assumed everywhere that each such function is Lipschitz continuous, and differentiable. Since in engineering optimization, the dimensionality of the design space is usually much higher than that of its image in the objective function space ($n \gg m$), it will be assumed hereafter that $n > m$.

Additionally, it will be assumed in this treatment, that the scalar objectives $f_j(\cdot)$ in $\mathbf{f}(\cdot)$ are *quasiconcave* functions. Quasiconcavity generalizes the notion of concavity. Any given function $f(\cdot)$ is (strictly) quasiconcave if and only if for every $\vartheta \in \mathbb{R}$, the upper contour set $\{\mathbf{x} | f(\mathbf{x}) \geq \vartheta\}$ is (strictly) convex. Two sufficient conditions for quasiconcavity are stated in the following axiom.

Axiom-1. With $\mathbf{x}, \mathbf{y} \in \mathcal{F}_\mathbf{x} \subseteq \mathbb{R}^\mathrm{n}$ being any pair of vectors in the convex domain $\mathcal{F}_\mathbf{x}$, either of following inequality is a sufficient condition for the quasiconcavity of the function f(·),

$$\begin{cases} \theta \in [0,1] \Rightarrow f(\theta\mathbf{x} + (1-\theta)\mathbf{y}) \geq \min\big(f(\mathbf{x}), f(\mathbf{y})\big) = 0, \\ f(\mathbf{x}) \geq f(\mathbf{x}+\mathbf{y}) \Rightarrow \mathbf{y}^\mathrm{T}\nabla_\mathbf{x} f(\mathbf{x}) \geq 0. \end{cases} \tag{3.9}$$

The function's quasiconcavity becomes strict if all inequalities are strict in the first condition in Eqn. (3.9). In addition to being sufficient, this condition is also a necessary one for quasiconcavity. It may be noted that the second condition in Eqn. (3.9) defines pseudoconcave

functions that is outside the scope of this research, and is therefore a stricter requirement than the first. Formal proofs can be found in [80], [81].

The relationship $\mathbf{f}(\mathbf{x}) > \mathbf{f}(\mathbf{y})$ is used to denote that $f_j(\mathbf{x}) > f_j(\mathbf{y}), \forall j$. Analogous elementwise interpretations apply to the remaining inequality relationships $\geq, <$, and $\leq$. Given two vectors $\mathbf{x}, \mathbf{y} \in \mathcal{F}_\mathbf{x}$, $\mathbf{x}$ *weakly dominates* $\mathbf{y}$ when $\mathbf{f}(\mathbf{x}) \geq \mathbf{f}(\mathbf{y})$. When there is at least some $f_j(\cdot)$ such than $f_j(\mathbf{x}) > f_j(\mathbf{y})$, then $\mathbf{x}$ *dominates* $\mathbf{y}$. This (weak) dominance relationship is denoted as $\mathbf{x} \succ \mathbf{y}$ ($\mathbf{x} \succcurlyeq \mathbf{y}$). Any point[1] $\mathbf{x} \in \mathcal{F}_\mathbf{x}$ is *locally* (*weakly*) *Pareto-optimal* if and only if there exists a quantity $\sigma > 0$ satisfying the condition,

$$\mathbf{y} \in \mathcal{F}_\mathbf{x} \cap \mathcal{B}(\mathbf{x}, \sigma) \Rightarrow \mathbf{x} \succ \mathbf{y} \ (\mathbf{x} \succcurlyeq \mathbf{y}). \tag{3.10}$$

In the above expression, $\mathcal{B}(\mathbf{x}, \sigma) \subset \mathbb{R}^n$ is a ball centered around $\mathbf{x}$ with radius σ, $\mathcal{B}(\mathbf{x}, \sigma) = \{\mathbf{y} \in \mathbb{R}^n | \|\mathbf{y} - \mathbf{x}\| > \sigma\}$. If the condition in Eqn. (3.10) holds in the limiting case $\sigma \to \infty$, then $\mathbf{x}$ is said to be (*weakly*) *Pareto-optimal*. The image of the set of all Pareto-optimal points is the CVOP's Pareto front.

From here onwards, the $n \times m$ Jacobian matrix will be denoted as $\nabla_\mathbf{x}\mathbf{f}(\mathbf{x}) \triangleq [\nabla_\mathbf{x} f_1(\mathbf{x}) \ \cdots \ \nabla_\mathbf{x} f_M(\mathbf{x})]$. For simplicity it is assumed to be of full column rank unless noted otherwise. There is a useful relationship between the gradient vectors $\nabla_\mathbf{x} f_j(\mathbf{x})$ of the $j \in \{1,2,\ldots,m\}$ objectives of a locally Pareto optimal point $\mathbf{x}$. Consider another point, $\mathbf{y} = \mathbf{x} + \delta\mathbf{x} \in \mathcal{F}_\mathbf{x}$, where $\delta\mathbf{x}$ is an infinitesimal perturbation of $\mathbf{x}$ so that higher order terms in the Taylor's series expansion can be ignored. Hence, in the limiting case of $\delta\mathbf{x} \to \mathbf{0}$, $\mathbf{f}(\mathbf{y}) = \mathbf{f}(\mathbf{x}) + \nabla_\mathbf{x}^\mathrm{T}\mathbf{f}(\mathbf{x})\delta\mathbf{x}$. From (10), $\mathbf{f}(\mathbf{x}) > \mathbf{f}(\mathbf{y})$, so that $\nabla_\mathbf{x}^\mathrm{T}\mathbf{f}(\mathbf{x})\delta\mathbf{x} < \mathbf{0}$. Suppose $\delta\mathbf{x}$ is chosen such that all of its components are positive ($\delta\mathbf{x} > \mathbf{0}$). In other words, for every $f_j(\cdot)$ there must be at least one function $f_i(\cdot)$ such that $\nabla_\mathbf{x}^\mathrm{T} f_i(\mathbf{x})\delta\mathbf{x}$ and $\nabla_\mathbf{x}^\mathrm{T} f_j(\mathbf{x})\delta\mathbf{x}$ have opposite signs. This observation is significant. If such a $\delta\mathbf{x}$ does not exist, then from the convexity of $\mathcal{F}_\mathbf{x}$, another vector $\delta\mathbf{x} < \mathbf{0}$ exists, leading to the same observation. Another way of interpreting this observation is that from a Pareto optimal point any improvement (i.e. increase) in one objective can only be accomplished at the expense of another.

In CVOPs, there exist necessary and sufficient conditions that are analogous to the KKT conditions in scalar constrained optimization. These are the *Fritz-John* (FJ) *conditions* [77], [80], [82], for local Pareto optimality. We state these conditions in the following axiom.

[1] The terms 'point' and 'vector' are used interchangeably.

Axiom-2. The vector $\mathbf{x} \in \mathcal{F}_{\mathbf{x}}$ is locally Pareto optimal if there exist vectors $\boldsymbol{\xi} \in \mathbb{R}^m$, $\boldsymbol{\lambda} \in \mathbb{R}^p$, and $\boldsymbol{\mu} \in \mathbb{R}^q$ satisfying the following conditions,

$$\begin{cases} \boldsymbol{\xi} \geq \mathbf{0}_m;\ \boldsymbol{\lambda} \geq \mathbf{0}_p \\ \boldsymbol{\lambda}^{\mathrm{T}}(\mathbf{A}^{\mathrm{T}}\mathbf{x} + \mathbf{a}) = 0; \\ \nabla_{\mathbf{x}}\mathbf{f}(\mathbf{x})\boldsymbol{\xi} - \mathbf{A}\boldsymbol{\lambda} - \mathbf{B}\boldsymbol{\mu} = \mathbf{0}_p. \end{cases} \tag{3.11}$$

The Fritz-John conditions reduce to the well-known first order KKT optimality conditions with $\boldsymbol{\xi} \neq \mathbf{0}_m$, the weighted sum of the objectives, $\boldsymbol{\xi}^{\mathrm{T}}\mathbf{f}(\mathbf{x})$ acting as the equivalent scalar objective and treating the function $\mathfrak{L}(\mathbf{x}, \boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \boldsymbol{\xi}^{\mathrm{T}}\mathbf{f}(\mathbf{x}) - \boldsymbol{\lambda}^{\mathrm{T}}(\mathbf{A}^{\mathrm{T}}\mathbf{x} + \mathbf{a}) - \boldsymbol{\mu}^{\mathrm{T}}(\mathbf{B}^{\mathrm{T}}\mathbf{x} + \mathbf{b})$ as the equivalent Lagrangian function, as seen in [32].

In multi-gradient ascent algorithms, the *common* (*feasible*) *ascent direction* is a vector $\boldsymbol{\omega}$ such that for some $\delta > 0$, $\mathbf{x} + \delta\boldsymbol{\omega} \succ \mathbf{x}$. Multi-gradient ascent involves iterative increments of $\mathbf{x}$ along common ascent directions. Using Taylor's series expansion it can readily be shown that any common ascent direction $\boldsymbol{\omega}$ must be expressed as a convex combination of the gradients,

$$\boldsymbol{\omega} = \nabla_{\mathbf{x}}\mathbf{f}(\mathbf{x})\boldsymbol{\xi}, \tag{3.12}$$

where $\boldsymbol{\xi} > \mathbf{0}$. In MGDA, the weights in $\boldsymbol{\xi}$ of the gradients $\nabla_{\mathbf{x}} f_j(\mathbf{x})$ are constrained so that $\mathbf{1}_m^{\mathrm{T}}\boldsymbol{\xi} = 1$. The direction $\boldsymbol{\omega}$ is chosen to be the minimum norm element in the convex hull of the gradients.

Axiom-3 below, stems from the observation made earlier that at a Pareto-optimal point, any gain with respect to an objective will always be at the expense of another. Formal proofs can be found in [29], [30], [31].

Axiom-3. At any point $\mathbf{x} \in \mathcal{F}_{\mathbf{x}}$, if no common feasible ascent direction satisfying Eqn. (3.12) exists, then $\mathbf{x}$ is locally Pareto optimal.

3.3.2 Augmented Lagrangian Multi-Gradient Ascent

Since $\boldsymbol{\omega}$ in Eqn. (3.12) will be used to increment $\mathbf{x} \in \mathcal{F}_{\mathbf{x}}$, we must have $\mathbf{x} + \boldsymbol{\omega} \in \mathcal{F}_{\mathbf{x}}$, so that, $\mathbf{A}^{\mathrm{T}}(\mathbf{x} + \boldsymbol{\omega}) + \mathbf{a} \leq \mathbf{0}$, and, $\mathbf{B}^{\mathrm{T}}(\mathbf{x} + \boldsymbol{\omega}) + \mathbf{b} = \mathbf{0}$. A *sufficient* condition on $\boldsymbol{\omega}$ to satisfy the above constraints would be that $\mathbf{A}^{\mathrm{T}}\boldsymbol{\omega} \leq \mathbf{0}$, $\mathbf{B}^{\mathrm{T}}\boldsymbol{\omega} = \mathbf{0}$. ALMA does not aim to bring the point $\mathbf{x}$ to the feasible region. This goal can be achieved separately through any other constrained optimization

algorithm. The goal is merely that when ALMA increment $\mathbf{x}$ by $\boldsymbol{\omega}$ should not violate the constraint any further. Inequality and equality constraints in Figure 3.4 are considered separately below.

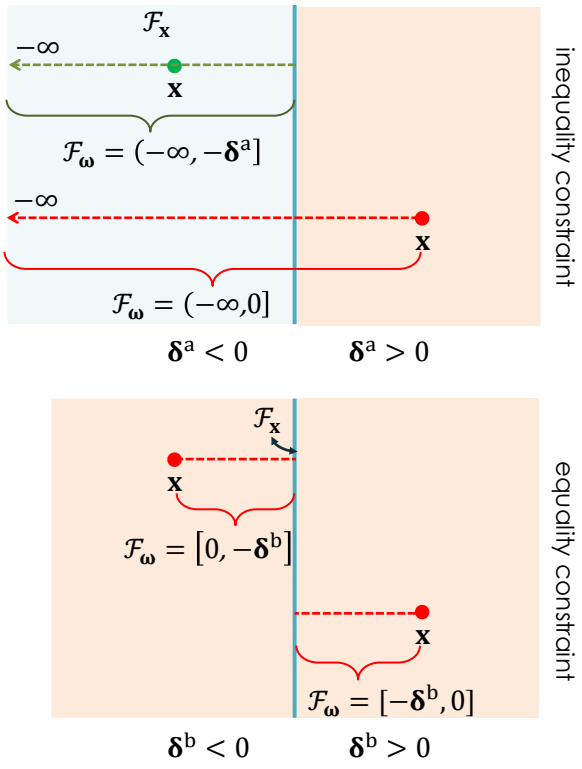

Figure 3.4: Feasible regions of ω.

(i) Suppose the inequality constraint is inactive so that $\mathbf{A}^{\mathrm{T}}\mathbf{x} + \mathbf{a} = \boldsymbol{\delta}^{\mathrm{a}} < \mathbf{0}$. In Figure 3.4 (top), this corresponds to the small, green circle representing $\mathbf{x}$. After replacing $\mathbf{x}$ with $\mathbf{x} + \boldsymbol{\omega}$, the constraint must not be violated. The analogous condition on $\mathbf{x} + \boldsymbol{\omega}$ is, $\mathbf{A}^{\mathrm{T}}(\mathbf{x} + \boldsymbol{\omega}) + \mathbf{a} \leq \mathbf{0}$, which upon simplification yields, $\mathbf{A}^{\mathrm{T}}\boldsymbol{\omega} \leq -\boldsymbol{\delta}^{\mathrm{a}}$. Next, suppose the inequality constraint is violated so that $\mathbf{A}^{\mathrm{T}}\mathbf{x} + \mathbf{a} = \boldsymbol{\delta}^{\mathrm{a}} > \mathbf{0}$, with such an $\mathbf{x}$ shown as a red circle in Figure 3.4 (top). For ALMA not to move $\mathbf{x} + \boldsymbol{\omega}$ further away from the feasible region than $\mathbf{x}$, we must have $\mathbf{A}^{\mathrm{T}}(\mathbf{x} + \boldsymbol{\omega}) + \mathbf{a} \leq \boldsymbol{\delta}^{\mathrm{a}}$. In other words, $\mathbf{A}^{\mathrm{T}}\boldsymbol{\omega} \leq \mathbf{0}$. Combining both the cases, the constraint upon the direction, $\boldsymbol{\omega}$ must be $\mathbf{A}^{\mathrm{T}}\boldsymbol{\omega} \leq -[\boldsymbol{\delta}^{\mathrm{a}}]_{-}$. Here, $[\boldsymbol{\delta}]_{-} = \min(\boldsymbol{\delta}, \mathbf{0})$, with the minimization being carried out in a component-wise manner. Similarly, $[\boldsymbol{\delta}]_{+} = \max(\boldsymbol{\delta}, \mathbf{0})$.

(ii) Suppose the equality constraint is violated in the positive direction so that $\mathbf{B}^{\mathrm{T}}\mathbf{x} + \mathbf{b} = \boldsymbol{\delta}^{\mathrm{b}} > \mathbf{0}$. Figure 3.4 (bottom) depicts such a situation where the red circle is to

the right of $\mathcal{F}_{\mathbf{x}}$ (vertical line). As we chose to ensure that the updated variable $\mathbf{x}+\boldsymbol{\omega}$ not move any further away from the feasible region, we must have, $\mathbf{0} \leq \mathbf{B}^{\mathrm{T}}(\mathbf{x}+\boldsymbol{\omega})+\mathbf{b} \leq \boldsymbol{\delta}^{\mathrm{b}}$. This leads to the bounds, $-\boldsymbol{\delta}^{\mathrm{b}} \leq \mathbf{B}^{\mathrm{T}}\boldsymbol{\omega} \leq \mathbf{0}$. Next, suppose the equality constraint is violated in the negative direction. In this case, $\mathbf{B}^{\mathrm{T}}\mathbf{x}+\mathbf{b} = \boldsymbol{\delta}^{\mathrm{b}} < \mathbf{0}$ yielding the bounds, $\mathbf{0} \leq \mathbf{B}^{\mathrm{T}}\boldsymbol{\omega} \leq -\boldsymbol{\delta}^{\mathrm{b}}$. Combining both cases together, the equivalent condition that $\mathbf{x}+\boldsymbol{\omega}$ is no further away from $\mathcal{F}_{\mathbf{x}}$ than $\mathbf{x}$, is given by, $-\left[\boldsymbol{\delta}^{\mathrm{b}}\right]_{+} \leq \mathbf{B}^{\mathrm{T}}\boldsymbol{\omega} \leq -\left[\boldsymbol{\delta}^{\mathrm{b}}\right]_{-}$.

Combining the observations for both kinds of constraints allows us to define feasible region $\mathcal{F}_{\boldsymbol{\omega}}$ for the direction $\boldsymbol{\omega}$ in the following manner,

$$\mathcal{F}_{\boldsymbol{\omega}} \triangleq \left\{ \boldsymbol{\omega} \middle| \begin{array}{c} \mathbf{A}^{\mathrm{T}}\boldsymbol{\omega} \leq -[\boldsymbol{\delta}^{\mathrm{a}}]_{-}, \\ -\left[\boldsymbol{\delta}^{\mathrm{b}}\right]_{+} \leq \mathbf{B}^{\mathrm{T}}\boldsymbol{\omega} \leq -\left[\boldsymbol{\delta}^{\mathrm{b}}\right]_{-} \end{array} \right\}, \tag{3.13}$$

where $\boldsymbol{\delta}^{\mathrm{a}} = \mathbf{A}^{\mathrm{T}}\mathbf{x}+\mathbf{a}$, and $\boldsymbol{\delta}^{\mathrm{b}} = \mathbf{B}^{\mathrm{T}}\mathbf{x}+\mathbf{b}$.

The term $\nabla_{\mathbf{x}}\mathbf{f}(\mathbf{x})\boldsymbol{\xi}$ with $\boldsymbol{\xi} \geq \mathbf{0}_m, \mathbf{1}_m^{\mathrm{T}}\boldsymbol{\xi} = 1$ may not guarantee that the incremented $\mathbf{x}$ remains in the feasible region. In ALMA the increment $\boldsymbol{\omega}$ on $\mathbf{x}$ is a fraction $\nu \in (0,1]$ of that obtained by MGDA such that $\boldsymbol{\omega} \in \mathcal{F}_{\boldsymbol{\omega}}$. Furthermore, ν should be maximized so that the increment is as close to $\nabla_{\mathbf{x}}\mathbf{f}(\mathbf{x})\boldsymbol{\xi}$ as possible. This leads to the following bilevel problem,

$$\boldsymbol{\omega} = \nu\nabla_{\mathbf{x}}\mathbf{f}(\mathbf{x})\boldsymbol{\xi}, \tag{3.14}$$

where,

$$\boldsymbol{\xi}, \nu = \underset{\substack{0\leq\nu\leq 1, \nu\boldsymbol{\xi}^{\mathrm{T}}\nabla_{\mathbf{x}}\mathbf{f} \in \mathcal{F}_{\boldsymbol{\omega}} \\ \boldsymbol{\xi} = \underset{\boldsymbol{\xi}\geq\mathbf{0},\ \mathbf{1}^{\mathrm{T}}\boldsymbol{\xi}=1}{\operatorname{argmin}} \|\nabla_{\mathbf{x}}\mathbf{f}(\mathbf{x})\boldsymbol{\xi}\|}}{\operatorname{argmax}} \nu. \tag{3.15}$$

This scheme is illustrated in Figure 3.5 for a bi-objective CVOP. The shaded elliptical region represents $\mathcal{F}_{\mathbf{x}}$. The vector $\nabla_{\mathbf{x}}\mathbf{f}(\mathbf{x})\boldsymbol{\xi}$ (orange dotted arrow) is the perpendicular bisector of the shaded triangle shaped region whose sides are $\nabla_{\mathbf{x}}f_1(\mathbf{x})$ and $\nabla_{\mathbf{x}}f_2(\mathbf{x})$ (red, dotted arrows). The increment $\boldsymbol{\omega}$ is the solid green arrow in Figure 3.5.

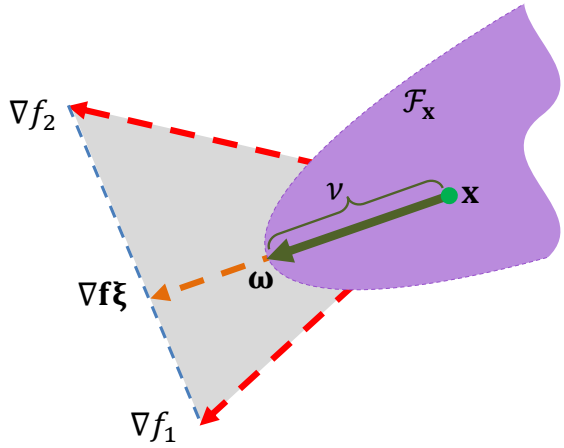

Figure 3.5: Common feasible ascent direction

Suppose $\mathbf{x}$ is infeasible – a situation that occurs commonly in exterior point algorithms such as the augmented Lagrangian method that ALMA incorporates. Unless the point $\mathbf{x}$ is sufficiently close to $\mathcal{F}_{\mathbf{x}}$, there may not exist any $\nu \in [0,1]$ such that $\boldsymbol{\omega} \in \mathcal{F}_{\boldsymbol{\omega}}$. As a result, ALMA does not increment $\mathbf{x}$ in the direction of the gradients until it is either inside $\mathcal{F}_{\mathbf{x}}$ or close enough to it. The point is still updated using the terms involving the dual variables. This is a desirable feature as it helps $\mathbf{x}$ move quicker towards the feasible region while allowing the dual variables acquire more consistent values.

3.3.3 DSO Level Pareto-Optimality

It is now shown that any locally Pareto optimal point obtained by ALMA is Pareto optimal.

Theorem 2. *Jain's index* $J(\cdot)$ *is strictly quasiconcave in* $\mathbb{R}^n$.

Proof: Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ be two independent non-zero vectors such that $f(\mathbf{x}) \geq f(\mathbf{x}+\mathbf{y})$. For a function $f(\cdot)$ to be quasiconcave, $\mathbf{y}^{\mathrm{T}}\nabla_{\mathbf{x}} f(\mathbf{x}) \geq 0$. Suppose $\mathbf{x}, \mathbf{y}$ are such that $f(\mathbf{x}) \geq f(\mathbf{x}+\mathbf{y})$ and $\mathbf{y}^{\mathrm{T}}\nabla_{\mathbf{x}} f(\mathbf{x}) = 0$. From the generalized mean value theorem, there must exist a $\rho \in [0,1]$, such that, $f(\mathbf{x}+\mathbf{y}) = f(\mathbf{x}) + \mathbf{y}^{\mathrm{T}}\nabla_{\mathbf{x}} f(\mathbf{x}) + \frac{1}{2}\mathbf{y}^{\mathrm{T}}\nabla_{\mathbf{x}}^2 f\big(\mathbf{x}+\rho(\mathbf{y}-\mathbf{x})\big)\mathbf{y}$. As $f(\mathbf{x}) \geq f(\mathbf{x}+\mathbf{y})$ and $\mathbf{y}^{\mathrm{T}}\nabla_{\mathbf{x}} f(\mathbf{x}) = 0$, it must be true that the third term, $\mathbf{y}^{\mathrm{T}}\nabla_{\mathbf{x}}^2 f\big(\mathbf{x}+\rho(\mathbf{y}-\mathbf{x})\big)\mathbf{y} \leq 0$. Letting $\rho = 0$, a sufficient condition for $f(\cdot)$ is that if $\mathbf{y}^{\mathrm{T}}\nabla_{\mathbf{x}} f(\mathbf{x}) \geq 0$ for some $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, then $\mathbf{y}^{\mathrm{T}}\nabla_{\mathbf{x}}^2 f(\mathbf{x})\mathbf{y} \leq 0$.

From Eqn. (3.5) it can be shown that,

$$\nabla_{\mathbf{x}} J(\mathbf{x}) = 2\sqrt{J(\mathbf{x})}\left(\frac{\mathbf{1}}{\|\mathbf{1}\|\|\mathbf{x}\|} - \sqrt{J(\mathbf{x})}\frac{\mathbf{x}}{\|\mathbf{x}\|^2}\right). \tag{3.15}$$

The Hessian of $J(\mathbf{x})$ can be obtained easily by differentiating the above expression,

$$\begin{aligned}\nabla_{\mathbf{x}}^2 J(\mathbf{x})\mathbf{y} = &-\frac{2}{\|\mathbf{x}\|^2}\sqrt{J(\mathbf{x})}\left(\frac{\mathbf{1}\mathbf{x}^{\mathrm{T}}}{\|\mathbf{1}\|\|\mathbf{x}\|} - \sqrt{J(\mathbf{x})}\frac{\mathbf{x}\mathbf{x}^{\mathrm{T}}}{\|\mathbf{x}\|^2}\right)\\ &-\frac{2}{\|\mathbf{x}\|^2}\left(\frac{-\mathbf{1}}{\|\mathbf{1}\|} + 2\frac{\mathbf{x}}{\|\mathbf{x}\|}\sqrt{J(\mathbf{x})}\right)\left(\frac{\mathbf{1}^{\mathrm{T}}}{\|\mathbf{1}\|} - \sqrt{J(\mathbf{x})}\frac{\mathbf{x}^{\mathrm{T}}}{\|\mathbf{x}\|}\right)\\ &-\frac{1}{\|\mathbf{x}\|^2}J(\mathbf{x})\left(2\mathbf{I} - \frac{\mathbf{x}\mathbf{x}^{\mathrm{T}}}{\|\mathbf{x}\|^2}\right).\end{aligned} \tag{3.17}$$

Rearranging terms and simplifying further using Eqn. (3.15) leads to,

$$\nabla_{\mathbf{x}}^2 J(\mathbf{x}) = -\frac{1}{\|\mathbf{x}\|^2}\nabla_{\mathbf{x}} J(\mathbf{x})\mathbf{x}^{\mathrm{T}} + \left(\frac{1}{2J(\mathbf{x})}\nabla_{\mathbf{x}} J(\mathbf{x}) - \frac{\mathbf{x}}{\|\mathbf{x}\|^2}\right)\nabla_{\mathbf{x}}^{\mathrm{T}} J(\mathbf{x}) - \frac{1}{\|\mathbf{x}\|^2}J(\mathbf{x})\left(2\mathbf{I} - \frac{\mathbf{x}\mathbf{x}^{\mathrm{T}}}{\|\mathbf{x}\|^2}\right). \tag{3.18}$$

Using the above expression for the Hessian in Eqn. (3.17),

$$\begin{aligned}\mathbf{y}^{\mathrm{T}}\nabla_{\mathbf{x}}^2 J(\mathbf{x})\mathbf{y} = &\frac{1}{\|\mathbf{x}\|^2}\mathbf{y}^{\mathrm{T}}\nabla_{\mathbf{x}} J(\mathbf{x})\mathbf{x}^{\mathrm{T}}\mathbf{y} + \mathbf{y}^{\mathrm{T}}\left(\frac{1}{2J(\mathbf{x})}\nabla_{\mathbf{x}} J(\mathbf{x}) - \frac{\mathbf{x}}{\|\mathbf{x}\|^2}\right)\nabla_{\mathbf{x}}^{\mathrm{T}} J(\mathbf{x})\mathbf{y}\\ &-\frac{\mathbf{y}^{\mathrm{T}}}{\|\mathbf{x}\|^2}J(\mathbf{x})\left(2\mathbf{I} - \frac{\mathbf{x}\mathbf{x}^{\mathrm{T}}}{\|\mathbf{x}\|^2}\right)\mathbf{y}.\end{aligned} \tag{3.19}$$

Under the assumption that $\mathbf{y}^{\mathrm{T}}\nabla_{\mathbf{x}} J(\mathbf{x}) = 0$, the above equality in Eqn. (3.19) can be simplified to,

$$\mathbf{y}^{\mathrm{T}}\nabla_{\mathbf{x}}^2 J(\mathbf{x})\mathbf{y} = -\frac{1}{\|\mathbf{x}\|^4}J(\mathbf{x})(2\|\mathbf{x}\|^2\|\mathbf{y}\|^2 - (\mathbf{x}^{\mathrm{T}}\mathbf{y})^2). \tag{3.20}$$

From the Cauchy-Schwarz inequality, as $\mathbf{x}$ and $\mathbf{y}$ are independent non-zero vectors, $|\mathbf{x}^{\mathrm{T}}\mathbf{y}| < \|\mathbf{x}\|\|\mathbf{y}\|$ so that $\mathbf{y}^{\mathrm{T}}\nabla_{\mathbf{x}}^2 J(\mathbf{x})\mathbf{y} < 0$. This proves the strict quasiconcavity of $J(\mathbf{x})$.

■

Theorem-3 below provides sufficient conditions for the convergence of ALMA towards Pareto optimal allocations.

Theorem 3. *If all scalar objectives $f_j(\cdot)$ of the vector function $\boldsymbol{f}(\cdot)$ are strictly quasiconcave, then any locally Pareto optimal point $\boldsymbol{x} \in \mathcal{F}_{\boldsymbol{x}}$ is globally Pareto optimal.*

Proof: Let $\mathbf{x}$ be a locally Pareto optimal point. Thus there exists a $\sigma > 0$ such that $\mathbf{x}$ dominates every other feasible point in the ball $\mathcal{B}(\mathbf{x}, \sigma)$. Assume that, contrary to the statement of this theorem, $\mathbf{x}$ is not globally Pareto optimal. Under these circumstances we pick an arbitrary point $\mathbf{y} \in \mathcal{F}_{\mathbf{x}}$ with $\|\mathbf{x} - \mathbf{y}\| > \sigma$, such that $\mathbf{x} \not\succcurlyeq \mathbf{y}$. In other words, there is an objective $f_j(\cdot)$ such that $f_j(\mathbf{y}) > f_j(\mathbf{x})$. From Eqn. (3.8), the feasible region $\mathcal{F}_{\mathbf{x}}$ is convex, so that for all $\theta \in [0,1]$, the point $\mathbf{z} = \theta\mathbf{x} + (1-\theta)\mathbf{y}$ must be feasible, i.e. $\mathbf{z} \in \mathcal{F}_{\mathbf{x}}$. From Eqn. (3.9) the strict quasiconcavity of $f_j(\cdot)$ implies that $f_j(\mathbf{z}) > \min(f_j(\mathbf{x}), f_j(\mathbf{y}))$, i.e. $f_j(\mathbf{z}) > f_j(\mathbf{x})$. If θ is confined to the smaller interval, $(0, \|\mathbf{x} - \mathbf{y}\|^{-1}\sigma] \subset [0,1]$, then $\mathbf{z}$ lies inside the ball $\mathcal{B}(\mathbf{x}, \sigma)$. Since $\mathbf{z} \in \mathcal{B}(\mathbf{x}, \sigma)$, $\mathbf{x} \succcurlyeq \mathbf{z}$ so that $f_j(\mathbf{z}) \leq f_j(\mathbf{x})$, contradicting our previous observation that $f_j(\mathbf{z}) > f_j(\mathbf{x})$.

■

Theorem-3 is of significance to the DSO CVOP defined in section 3.2.4. Since it has been assumed that all agents' utility functions are strictly quasiconcave, consequently, $\mathcal{U}_k(\cdot)$, which is their sum over disjoint arguments, is also strictly quasiconcave. By analogous reasoning, so is the welfare function $\mathcal{W}(\cdot)$. In Theorem 2, the Jain's index is shown to be strictly quasiconcave; therefore the fairness measure, $\mathcal{R}(\cdot)$ is also strictly quasiconcave. In other words, all scalar components of $\boldsymbol{\Omega}(\cdot)$ in Eqn. (3.7) are strictly quasiconcave functions, whence by Theorem 3 the locally Pareto optimal point obtained by ALMA is Pareto optimal. More formal proofs of the theorem can be found in [77], [82].

3.3.4 DSO Algorithm

The DSO algorithm is a specific implementation of ALMA for energy allocation in distribution systems. In step 1, the aggregator receives the unit cost $\mathbf{c}$ from aggregator auctions, which is equal to $\nabla_{\mathbf{p}}\mathcal{W}(\mathbf{p})$ (Theorem-1). The other gradient, $\nabla_{\mathbf{p}}\mathcal{R}(\mathbf{p})$ is computed in step 2 where $\mathbf{p}$ is the value from the previous iteration. Using the constraint gaps $\boldsymbol{\delta}^*$ (step 3) the quantities involved in $\mathcal{F}_{\mathbf{x}}$ in Eqn. (3.8) are determined (step 4). Steps 5–8 implement (12) as follows. Steps 5 and 6 implement ascent direction as in MGDA. This is scaled by the factor ν so that increments do not produce

infeasible solutions. The dual variables are incremented (step 9) in accordance with augmented Lagrangian method (see [79] for details), following which the energy allocation is incremented (step 10), and returned to the aggregator (step 11) for the next round of aggregator auctions until convergence towards a locally Pareto optimum, which Theorem-3 shows to be Pareto optimal.

The steps involved in the DSO algorithm are outlined in Algorithm 2.

Algorithm 3.2: DSO Algorithm

until (termination) **do**

1. Receive $\mathbf{c}$ from aggregators
2. $\mathbf{g} \leftarrow \nabla_{\mathbf{p}} \mathcal{R}(\mathbf{p})$
3. $\underline{\boldsymbol{\delta}}^{V} \leftarrow -\mathbf{C}^{V}\mathbf{p} + \mathbf{c}_{l}^{V}, \qquad \overline{\boldsymbol{\delta}}^{V} \leftarrow \mathbf{C}^{V}\mathbf{p} + \mathbf{c}_{u}^{V}$

   $\boldsymbol{\delta}^{S} \leftarrow \mathbf{C}^{S}\mathbf{p} + \mathbf{c}_{0}^{S}, \boldsymbol{\delta}^{P_0} \leftarrow \boldsymbol{C}^{P_0}\mathbf{p} + c_{0}^{P_0} - P_0, \qquad \boldsymbol{\delta}^{\mathrm{B}} \leftarrow -\mathbf{c}^{\mathrm{T}}\mathbf{p} + c_0 P_0$
4. $\mathbf{A} \leftarrow \begin{bmatrix} -\mathbf{C}^{V} \\ \mathbf{C}^{V} \\ \mathbf{C}^{S} \\ -\mathbf{c}^{\mathrm{T}} \end{bmatrix}, \quad \mathbf{a} \leftarrow \begin{bmatrix} -\mathbf{c}_{l}^{V} \\ -\mathbf{c}_{u}^{V} \\ -\mathbf{c}_{0}^{S} \\ -c_0 P_0 \end{bmatrix}, \qquad \mathbf{B} \leftarrow \boldsymbol{C}^{P_0}, \qquad \mathbf{b} \leftarrow c_{0}^{P_0} - P_0.$
5. $\boldsymbol{\xi} \leftarrow \underset{\boldsymbol{\xi} \geq \mathbf{0}, \mathbf{1}^{\mathrm{T}}\boldsymbol{\xi}=1}{\operatorname{argmin}} \|\nabla_{\mathbf{x}} \mathbf{f}(\mathbf{x})\boldsymbol{\xi}\|$
6. $\boldsymbol{\omega}' \leftarrow \nabla_{\mathbf{x}} \mathbf{f}(\mathbf{x})\boldsymbol{\xi}$
7. $\nu \leftarrow \underset{\substack{0 \leq \nu \leq \nu_{\max} \\ \nu\boldsymbol{\omega}' \in \mathcal{F}_{\boldsymbol{\omega}}}}{\max} \nu$
8. $\boldsymbol{\omega} \leftarrow \nu\boldsymbol{\omega}'$
9. $\underline{\boldsymbol{\alpha}} \leftarrow \left[\underline{\boldsymbol{\alpha}} + \nu\eta_{k}^{V}\underline{\boldsymbol{\delta}}^{V}\right]_{+}, \quad \overline{\boldsymbol{\alpha}} \leftarrow \left[\overline{\boldsymbol{\alpha}} + \nu\eta_{k}^{V}\overline{\boldsymbol{\delta}}^{V}\right]_{+}, \ \boldsymbol{\beta} \leftarrow [\boldsymbol{\beta} + \nu\eta_{k}^{S}\boldsymbol{\delta}^{S}]_{+}$

   $\lambda \leftarrow \lambda + \nu\eta_{k}^{P_0}\delta^{P_0}, \qquad \gamma \leftarrow \left[\gamma + \nu\eta_{k}^{\mathrm{B}}\delta^{\mathrm{B}}\right]_{+}$
10. $\Delta\mathbf{p} \leftarrow \boldsymbol{\omega} + \mathbf{C}^{V\mathrm{T}}\underline{\boldsymbol{\alpha}} - \mathbf{C}^{V\mathrm{T}}\overline{\boldsymbol{\alpha}} - \mathbf{C}^{S\mathrm{T}}\boldsymbol{\beta} - \lambda\boldsymbol{C}^{P_0} + \gamma\mathbf{c}$

    $\mathbf{p} \leftarrow \mathbf{p} + \nu\eta_{k}^{P}\Delta\mathbf{p}$
11. Send p to aggregators
12. $k \leftarrow k + 1$

end

3.4 Simulation Results

The proposed approach use was implemented on a modified IEEE 37-bus system as shown in Figure 3.6. Nodes containing the 17 aggregators appear as larger blue circles, whereas the remaining nodes are red filled circles. For convenience, the aggregators are indexed separately (inset in Figure 3.6) The number of agents in each aggregator was generated randomly between $G_k = 9$ and $G_k = 25$, with aggregators A4, A6, A10, A12 having a higher number of prosumers. Some agents were equipped with some PV generation ($g_k^i > 0$). The agent parameters, a_k^i,b_k^i, and g_k^i (see Figure 3.2) were generated randomly. All simulations were performed in MATLAB.

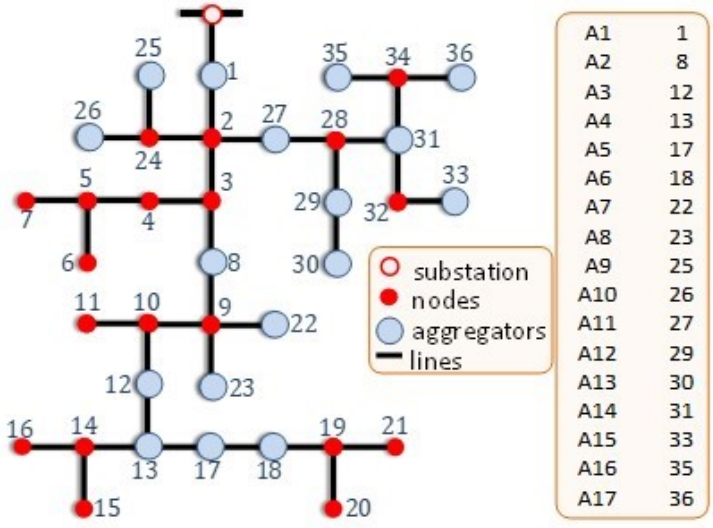

Figure 3.6: IEEE 37-bus system used as simulation platform

In order to see the effect of fairness, two simulations were done. The algorithm in [1] was implemented to obtain the efficient solution without any fairness. Following this, the DSO algorithm was implemented. Figure 3.7 compares the results of both simulations. The blue vertical bars are the power allocations of the aggregators (p_k) that were computed from the simulations without fairness. The aggregators' power allocations are shown as vertical bars that are colored blue (without fairness) and yellow (with fairness). The solid lines (without fairness) and dotted lines (with fairness) in the figure show the unit costs. The quantities with fairness are superscripted with asterisks (*).

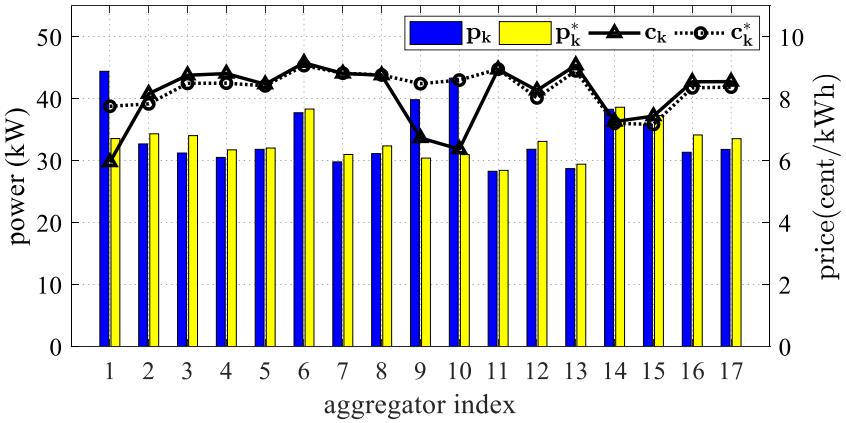

Figure 3.7: Aggregator power allocations and unit costs

From Figure 3.7 it can be seen that aggregators A1, A9, A10 received more power at lower unit cost in the absence of fairness. As can be seen in in Figure 3.6, these aggregators are positioned closer to the substation node (red circle with white interior). In contrast, aggregators A3, A12, A16, and A17 which are further away, experience higher unit cost and lower power allocation. The allocations obtained with fairness show how utilizing the Jain's index helps in mitigating this adverse effect. The fairness objective causes aggregators to be charged in a more equitable manner.

Figure 3.8 shows the progress of the algorithm with iteration. Figure 3.8 (top) shows how the inner product $\langle \mathcal{W}(\mathbf{p}), \mathcal{R}(\mathbf{p}) \rangle$ converges towards -1 at the Pareto front. The steadily increasing welfare (blue line) and fairness (red line) are shown in Figure 3.8 (bottom). Note that during the initial stages of the algorithm, the solution would be infeasible, explaining the initial fluctuations. Although 3,000 iterations were allowed for convergence, with very small random initialization of $\mathbf{p}$ and zero initial duals, with higher initial values, the algorithm would require as little as 500 iterations to converge, which is not much more than in [2] despite the presence of a vector objective.

Lastly, the tradeoff between welfare and fairness was investigated. Multiple simulations were carried out with random initialization of the primal and dual variables. Figure 3.9 depicts the resulting Pareto front.

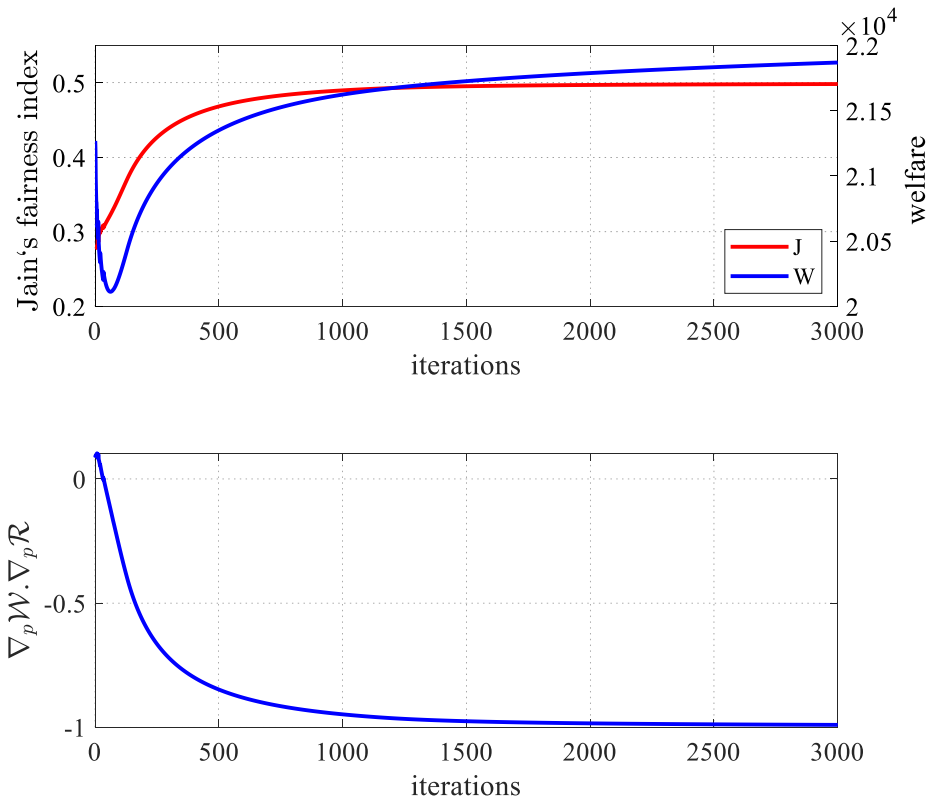

Figure 3.8: Welfare and Jain's index (top), inner product of gradients of welfare and fairness (bottom) vs. iteration

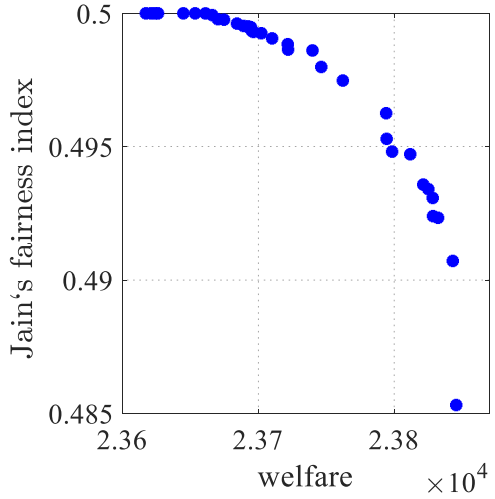

Figure 3.9: Jain's index vs. welfare

3.5 Conclusion

This research proposes a general-purpose approach for constrained vector optimization. The analytical treatment, albeit informal, shows that Pareto optimal solutions can be obtained as long as the objectives are strictly quasiconcave. This is an improvement over previous approaches, e.g. in, [5], [9], [71], [73], [83], [74], which routinely make more restrictive assumption of concave utilities. This is of significance as general econometric theory does not support prior assumptions of concavity, instead treating utilities justifiably as quasiconcave functions [78],[84],[85]. Efficiency-fairness tradeoff is a crucial issue in resource allocation [11], [65], [75],[86] with a significant amount of research using Jain's index (cf. [69], [79], [81]). In establishing the quasiconcavity of Jain's index, this research provides a theoretical justification for the application of vector optimization algorithms such as ALMA for tradeoff allocations.

It should be noted here that this research relies significantly on recent research. In particular, ALMA obtains the gradient in the same direction as in MGDA. The dual variables are incremented by the DSO algorithm in the same manner as in [79]. The aggregator auction is section 3.2.1 is a modification of those in [1], [73]. Additionally, there are a few limitations of this research that are outlined below.

It has been assumed that the feasibility constraints in Eqn. (3.8) were linear equality and inequality constraints. Although supported by Eqn. (3.3) for the energy grid, this assumption is over-simplistic for other applications. One possible improvement would be to linearize any nonlinear constraints at intermittent stages of the optimization algorithm. However, the effectiveness of ALMA in such situations needs to be further investigated.

The Pareto front in Figure 3.9 was obtained by randomly initializing the starting point. However, this approach highlights that ALMA can converge to any Pareto-optimal solution. Although the present simulation results indicated that any additional gain in efficiency was accompanied by a sharp drop in fairness, this may not necessarily be the case in other grids, and therefore is a potential limitation in ALMA. One option to exert more influence on the Pareto-optimal output allocation of ALMA, the authors suggest using MGDA's approach to deal with opposing objectives, which suggests the use of relaxation algorithms to converge towards generalized Nash equilibrium. Alternately, only one of the scalar functions in Eqn. (3.7) may be used as a scalar objective, while imposing bounds either on the other objective or on both, as

additional constraints. A similar constrained method has been used in [37]. Both approaches while remaining quite out of scope in this study, are worthwhile directions for further investigation.

In the absence of any secondary stage to navigate the Pareto front until a user-specified suitable point is reached, how useful is ALMA for use in the energy grid? Fortunately, in day-ahead planning where scheduling is usually done in an hourly manner, the allocation $\mathbf{p}_t$ during any hourly time interval t would not differ significantly from that of the previous interval $\mathbf{p}_{t-1}$. Initializing $\mathbf{p}_t$ to the previous $\mathbf{p}_{t-1}$, which is not only be feasible but also located very close to the desirable region in the Pareto front, would allow ALMA to converge to a Pareto optimal allocation at least an order of magnitude faster than what Figure 3.8 suggests. ALMA can be used in a similar fashion during real time operation, when actual user demands deviate from their forecasts. Barring unforeseen weather changes, as such deviations are usually very small, the planned value of $\mathbf{p}_t$ can readily be used as the initial point. Alternately, historical values from the DSO's database can also be adopted for initialization during weather related exigencies. Put together, these reasons largely obviate the need for ALMA to be equipped to move along the Pareto front.

The step sizes η_k^X of each dual variable X was obtained by the DSO algorithm as in [79]. However, energy allocations were incremented with η_k^P being kept proportional to $(1 + \langle \mathcal{W}(\mathbf{p}), \mathcal{R}(\mathbf{p}) \rangle)$. This method of stepwise updates made ALMA apply increasingly smaller increments as it approached the Pareto front. Although simulations in this research indicated its effectiveness, theoretical support for such a modification is lacking. The authors intend to extend this technique for more than two objectives, and to formally establish convergence limits with step sizes fashioned in this manner.

The proposed approach should be compared with novel algorithms for CVOP that were published recently in [35], [36], both of which appeared during a later phase of this research. In a similar manner, the effectiveness of ALMA with more than only two objectives, should be investigated in future research. The simulation results reported here serve as a proof-of-concept for a more general-purpose approach for large-scale constrained vector optimization.

Chapter 4 - An L0-Norm Constrained Non-Negative Matrix Factorization Algorithm for the Simultaneous Disaggregation of Fixed and Shiftable Loads

Energy disaggregation refers to the decomposition of energy use time series data into its constituent loads. This research decomposes daily use data of a household unit into fixed loads and one or more classes of shiftable loads. The latter are characterized by ON/OFF duty cycles. A novel algorithm based on non-negative matrix factorization (NMF) for energy disaggregation is proposed, where fixed loads are represented in terms of real-valued basis vectors, whereas shiftable loads are divided into binary signals. This binary decomposition approach directly applies L0-norm constraints on individual shiftable loads. The new approach obviates the need for more computationally intensive methods (e.g. spectral decomposition or mean-field annealing) that have been used in earlier research for these constraints. A probabilistic framework for the proposed approach has been addressed. The proposed approach's effectiveness has been demonstrated with real consumer energy data.

4.1 Introduction

Energy disaggregation refers to the decomposition of energy usage into multiple components in a physically meaningful way [49], [87]. For instance, daily energy consumption of a household unit can be disaggregated into various loads, such as refrigerator, air conditioner, lighting, pool pump, and other loads that are typically present in homes. In disaggregation tasks, the data samples are of the form of energy use over a fixed duration of time, at regularly spaced intervals. Until the last decade, energy disaggregation had met with little success. However, in recent years, NMF [88] has emerged as powerful tool for this purpose and has met with remarkable success.

The classical NMF algorithm decomposes an input data matrix $\mathbf{X}$ whose columns are $D \times 1$ sample vectors, into two factors, $\mathbf{W}$ and $\mathbf{H}$, so that their product equals $\mathbf{X}$. Usually, $\mathbf{X}$ has a very large number of columns, which are independent samples. Although there exists an abundance of classical matrix methods to factorize a given matrix in such a manner, in NMF there is the additional constraint that $\mathbf{W} \geq \mathbf{0}$ and $\mathbf{H} \geq \mathbf{0}$ [89],[90]. This non-negativity requirement placed on

both **W** and **H** render NMF suitable for many applications. For instance, NMF can be used to decompose image sequences or audio power spectra into factors **W** and **H** that can only have non-negative values, since neither pixel values nor power components can be negative. This is also the situation in the present research. The columns of **W**, which are relatively lesser in number, serve as basis vectors so that each sample $\mathbf{x}(n), n \in \mathcal{N}$ which are columns of **X** can be represented as a weighted combination of the bases, with the non-negative weights being the corresponding column vectors $\mathbf{h}(n)$ of **H**. In load disaggregation, the basis vectors may correspond to individual appliances [49], [87],.

Due to the non-negativity constraints, **X** is not exactly factorizable into **W** and **H**; whence the goal of NMF is to seek an approximate solution, so that $\mathbf{X} \approx \mathbf{WH}$. NMF is an ill-posed problem since if $\mathbf{X} \approx \mathbf{WH}$ is one solution, then so are other factorizations of the form $\mathbf{X} \approx \mathbf{WPP}^{\mathrm{T}}\mathbf{H}$ where **P** is any rotation matrix that preserves non-negativity, that can be considered to be other solutions.

In this research, a novel NMF algorithm for load disaggregation has been proposed. It uses the Frobenius norm as objective, although the approach is generalized enough to be extended to others. The novelty of this approach is the manner in which the load is divided into two classes, (*i*) *fixed loads*, and, (*ii*) *shiftable loads*. This is a fundamental distinction between the different appliances in a typical household that has not been hitherto considered.

Fixed loads are associated with appliances that are in continual use throughout the day. Examples of fixed loads include lighting and refrigerators. On the other hand, shiftable loads pertain to appliances that are used intermittently, such as washers, dryers, air-conditioners, and ovens. The latter class of loads are characterized by duty cycles, with typical temporal profiles. Our approach tries to exploit this feature in shiftable loads to obtain improved load disaggregation. Moreover, direct L_0 norm constraints are imposed on the number of duty cycles, separately for each shiftable load.

4.2 Framework

4.2.1 Load Models

Each input $\mathbf{x}(n)$ is a vector of D regularly spaced samples of energy usage of a single household unit over a 24-hour period. Therefore, the index n can be regarded as the energy use of the n^{th} day

in the sample set $\mathcal{N}$. The purpose of the proposed NMF algorithm is to express each sample in the following manner.

$$\mathbf{x}(n) \approx \sum_{k\in\mathcal{F}} h_k^f(n)\mathbf{w}_k^f + \sum_j \sum_{k\in\mathcal{S}_j} h_{j,k}^s(n)\mathbf{w}_{j,k}^s. \tag{4.1}$$

In Eqn. (4.1), $\mathcal{F}$ is the fixed load basis set and $\mathcal{S}_j$ is that of the j^{th} shiftable load. Each $D \times 1$ vector $\mathbf{w}_k^f$ is a fixed load basis vector k ($k \in \mathcal{F}$). Likewise, each $D \times 1$ $\mathbf{w}_{j,k}^s$ is a basis vector k ($k \in \mathcal{S}_j$) of the j^{th} shiftable load. In terms of basis matrices,

$$\mathbf{X} \approx \widetilde{\mathbf{X}} = \mathbf{W}^f\mathbf{H}^f + \sum_j \mathbf{W}_j^s\mathbf{H}_j^s \tag{4.2}$$

The quantities $\mathbf{W}^f$ and $\mathbf{W}_j^s$ are basis matrices of dimensionalities $D \times |\mathcal{F}|$ and $D \times |\mathcal{S}_j|$. The matrices $\mathbf{H}^f = [\mathbf{h}^f(n)]_{n\in\mathcal{N}}$ and $\mathbf{H}_j^s = [\mathbf{h}_j^s(n)]_{n\in\mathcal{N}}$ are $|\mathcal{F}| \times N$ and $|\mathcal{S}_j| \times N$ dimensional arrays of weights $h_k^f(n)$ and $h_{j,k}^s(n) \in \{0,1\}$. This decomposition is illustrated in Figure 4.1.

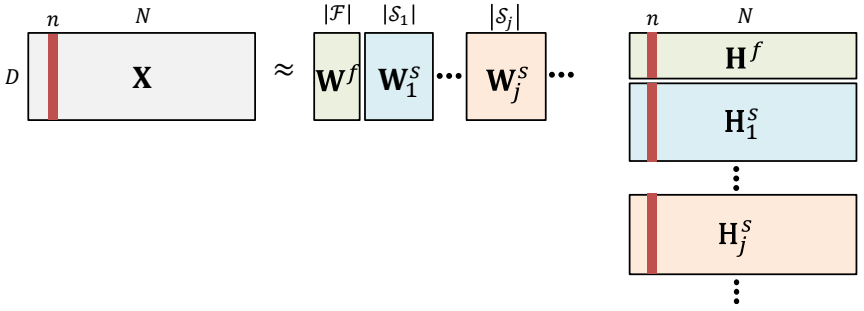

Figure 4.1: . Schematic of NMF factorization with separate basis and weight matrices for the fixed and shiftable loads.

The shiftable loads are characterized by OFF-ON duty cycles. It is assumed that during ON intervals, the j^{th} shiftable load draws a constant amount of energy p_j and that it can stay ON for a maximum of L_j time intervals. Hence,

$$\left\|\mathbf{h}_j^s(n)\right\|_0 \leq L_j. \tag{4.3}$$

Figure 4.2 shows the time profile of a shiftable load for the duration of a single day. It should be noted that as L_j is the upper limit on the number of duty cycles, the actual number of such cycles may be less than L_j.

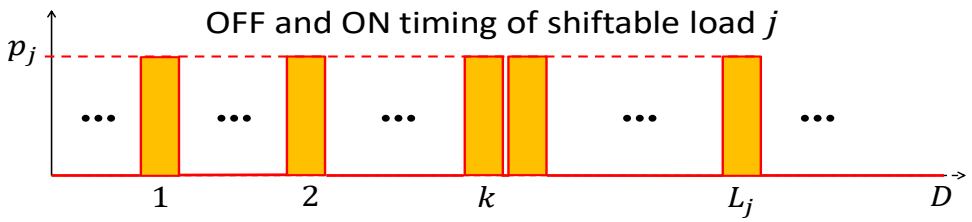

Figure 4.2: Illustrative usage profile of a shiftable load showing a maximum of L_j duty cycles, with each such cycle being a rectangular pulse of uniform peak p_j.

4.2.2 Objective

The squared Frobenius norm of the difference between the real data matrix $\mathbf{X}$ and its approximation $\widetilde{\mathbf{X}}$,

$$\Phi(\widetilde{\mathbf{X}}|\mathbf{X}) = \frac{1}{2}\left\|\mathbf{X} - \widetilde{\mathbf{X}}\right\|_F^2 = \frac{1}{2}\sum_n \|\mathbf{x}(n) - \tilde{\mathbf{x}}(n)\|_2^2 . \quad (4.4)$$

The expression for the approximation $\tilde{\mathbf{x}}(n)$ in Eqn.(4.4) has already been provided in Eqn. (4.1). The summation in Eqn. (4.4) is carried out over all samples in the set $\mathcal{N}$. A probabilistic justification for this choice is provided in section 4.3, which also serves as a theoretical basis for obtaining the weights associated with the shiftable loads in the proposed approach.

4.3 Proposed Approach

4.3.1 Probabilistic Framework

Consider the following probabilistic interpretation. The joint probability of $\mathbf{X}$, assuming that all samples are statistically *independent* is given by the following expression,

$$p\left[\mathbf{X}|\mathbf{W}^f, \mathbf{H}^f, \mathbf{W}_j^s, \mathbf{H}_j^s\right] = \prod_n p\left[\mathbf{x}(n)|\mathbf{W}^f, \mathbf{h}^f(n), \mathbf{W}_j^s, \mathbf{h}_j^s(n)\right]. \quad (4.5)$$

It is assumed that each sample $\mathbf{x}(n)$ follows a Gaussian distribution around its expected value $\tilde{\mathbf{x}}(n)$. In this case, the probability of each such sample can be expressed in the following manner,

$$p\left[\mathbf{x}(n)|\mathbf{W}^f, \mathbf{h}^f(n), \mathbf{W}_j^s, \mathbf{h}_j^s(n)\right] = \frac{1}{\sigma^D(2\pi)^{\frac{D}{2}}}\prod_d e^{-\frac{1}{2\sigma^2}\left(x_d(n)-\tilde{x}_d(n)\right)^2}. \tag{4.6}$$

The negated log probability of $\mathbf{x}(n)$ is,

$$\begin{aligned} -\log p\left[\mathbf{x}(n)|\mathbf{W}^f, \mathbf{h}^f(n), \mathbf{W}_j^s, \mathbf{h}_j^s(n)\right] &= \frac{1}{2\sigma^2}\sum_d \left(x_d(n)-\tilde{x}_d(n)\right)^2 + \log\sigma^D(2\pi)^{\frac{D}{2}} \\ &= \frac{1}{2\sigma^2}\left\|\mathbf{x}(n)-\tilde{\mathbf{x}}(n)\right\|_2^2 + \log\sigma^D(2\pi)^{\frac{D}{2}}. \end{aligned} \tag{4.7}$$

Hence the negative log probability of $\mathbf{X}$ is,

$$\begin{aligned} -\log p\left[\mathbf{X}|\mathbf{W}^f, \mathbf{H}^f, \mathbf{W}_j^s, \mathbf{H}_j^s\right] &= \prod_n p\left[\mathbf{x}(n)|\mathbf{W}^f, \mathbf{h}^f(n), \mathbf{W}_j^s, \mathbf{h}_j^s(n)\right] \\ &= \frac{1}{2\sigma^2}\left\|\mathbf{X}-\widetilde{\mathbf{X}}\right\|_F^2 + N\log\sigma^D(2\pi)^{\frac{D}{2}}, \end{aligned} \tag{4.8}$$

where the Frobenius norm of the difference between $\mathbf{X}$ and $\widetilde{\mathbf{X}}$ is given by the earlier expression in Eqn. (4.8).

Applying the maximum likelihood criterion to Eqn. (4.8),

$$\left[\mathbf{W}^f, \mathbf{H}^f, \mathbf{H}_j^s\right]_{ML} = \underset{\mathbf{W}^f, \mathbf{H}^f, \mathbf{H}_j^s}{\operatorname{arginf}} \frac{1}{2}\left\|\mathbf{X}-\widetilde{\mathbf{X}}\right\|_F^2. \tag{4.9}$$

The expression in Eqn. (4.9) above provides a theoretical justification for the choice of objective function in Eqn. (4.4) in the present approach.

The component of the objective function associated with the n^{th} sample is,

$$\begin{aligned}\varphi\big(\tilde{\mathbf{x}}(n)|\mathbf{x}(n)\big) &\triangleq \varphi(n)\\ &= \frac{1}{2}\|\mathbf{x}(n)-\tilde{\mathbf{x}}(n)\|_2^2\\ &= \frac{1}{2}\sum_d \big(x_d(n)-\tilde{x}_d(n)\big)^2.\end{aligned} \tag{4.10}$$

4.3.2 Multiplicative Update

Multiplicative updates are used for the matrices $\mathbf{W}^f$ and $\mathbf{H}^f$ that are associated with the fixed load. Although this method is quite routine in the existing literature on NMF, it is described here for the sake of completeness of this work. More details can be found in [49], [88], [89], [90].

Consider any parameter $\mathbf{P}$ (which can be either $\mathbf{W}^f$ or $\mathbf{H}^f$). Let $\nabla_{\mathbf{P}}\Phi$ be the gradient of tan arbitrary objective function Φ. Its gradient can be expressed in terms of its positive and negative components, so that $\nabla_{\mathbf{P}}\Phi = \nabla_{\mathbf{P}}^{+} - \nabla_{\mathbf{P}}^{-}$. In gradient descent, the update rule would have been of the form,

$$\mathbf{P} \leftarrow \mathbf{P} - \eta\nabla_{\mathbf{P}}^{+} + \eta\nabla_{\mathbf{P}}^{-}.$$

In the multiplicative method, the following multiplicative update replaces the gradient ascent step,

$$\mathbf{P} \leftarrow \mathbf{P} \circ \nabla_{\mathbf{P}}^{-} \oslash \nabla_{\mathbf{P}}^{+}.$$

In the proposed approach, the parameters $\mathbf{W}^f$ and $\mathbf{H}^f$ are subject to multiplicative updates. In order to do so, the derivatives of the objective must be first computed. It can be shown that,

$$\nabla_{\mathbf{W}^f}\frac{1}{2}\left\|\mathbf{X}-\widetilde{\mathbf{X}}\right\|_F^2 = -\mathbf{X}\circ\left(\mathbf{W}^f\mathbf{H}^f+\sum_j \mathbf{W}_j^s\mathbf{H}_j^s\right)^{\circ -1}\mathbf{H}^{f\,\mathrm{T}} + \mathbf{1}_{D\times N}\mathbf{H}^{f\,\mathrm{T}}. \tag{4.11}$$

$$\nabla_{\mathbf{H}^f}\frac{1}{2}\left\|\mathbf{X}-\widetilde{\mathbf{X}}\right\|_F^2 = -\mathbf{W}^{f\,\mathrm{T}}\mathbf{X}\circ\left(\mathbf{W}^f\mathbf{H}^f+\sum_j \mathbf{W}_j^s\mathbf{H}_j^s\right)^{\circ -1} + \mathbf{W}^{f\,\mathrm{T}}\mathbf{1}_{D\times N}. \tag{4.12}$$

Using the above expressions for the gradients, the update rules for the matrices pertaining to the fixed loads are,

$$\mathbf{W}^f \leftarrow \mathbf{W}^f \circ \frac{\left(\mathbf{X} \circ \widetilde{\mathbf{X}}^{\circ -1}\right)\mathbf{H}^{f^{\mathrm{T}}}}{\mathbf{1}_{D\times N}\mathbf{H}^{f^{\mathrm{T}}}} . \tag{4.13}$$

$$\mathbf{H}^f \leftarrow \mathbf{H}^f \circ \frac{\mathbf{W}^{f^{\mathrm{T}}}\left(\mathbf{X} \circ \widetilde{\mathbf{X}}^{\circ -1}\right)}{\mathbf{W}^{f^{\mathrm{T}}}\mathbf{1}_{D\times N}} . \tag{4.14}$$

Since the columns of $\mathbf{W}^f$ must be unit vectors, they are normalized at the end of each multiplicative update step as shown below,

$$\mathbf{w}_k^f \leftarrow \frac{\mathbf{w}_k^f}{\left\|\mathbf{w}_k^f\right\|}. \tag{4.15}$$

4.3.3. Sparsity Constrained Binary Updates

In order to train $\mathbf{H}_j^s$, $\operatorname{arginf}_{\mathbf{H}_j^s} \Phi$ is obtained using the binary updating heuristic proposed in this research. This heuristic directly imposes the L_0 norm constraint. The shiftable basis matrices $\mathbf{W}_j^s$ remain fixed throughout the training process.

It should be noted that since $h_{j,k}^s(n) \in \{0,1\}$ and $\left\|\mathbf{h}_j^s(n)\right\|_0 \leq L_j$, computing the optimal weight matrix $\mathbf{H}_j^s$ is an NP-hard problem. Therefore a hill-climbing heuristic is proposed. The component of the total error due to any sample n is given by the following expression,

$$\begin{aligned}
\mathbf{e}(n) &= \tilde{\mathbf{x}}(n) - \mathbf{x}(n) \\
&= \mathbf{W}^f\mathbf{h}^f(n) + \sum_j \mathbf{W}_j^s\mathbf{h}_j^s(n) - \mathbf{x}(n) \\
&= \mathbf{W}^f\mathbf{h}^f(n) + \mathbf{W}_j^s\mathbf{h}_j^s(n) + \sum_{j'\neq j} \mathbf{W}_{j'}^s\mathbf{h}_{j'}^s(n) - \mathbf{x}(n) \\
&= \mathbf{W}_j^s\mathbf{h}_j^s(n) - \mathbf{r}_j(n).
\end{aligned} \tag{4.16}$$

In the above equality, $\mathbf{r}_j(n)$ is the residual approximation of the n^{th} sample when ignoring the j^{th} shiftable load as shown below,

$$\mathbf{r}_j(n) = \mathbf{x}(n) - \mathbf{W}^f \mathbf{h}^f(n) - \sum_{j' \neq j} \mathbf{W}^s_{j'} \mathbf{h}^s_{j'}(n).$$

$$\approx \sum_{k \in \mathcal{S}_j} \mathbf{w}^s_{j,k} h^s_{j,k}(n). \tag{4.17}$$

The objective function for any given sample n provided earlier in Eqn. (4.10) can be expressed as given below,

$$\varphi(n) = \frac{1}{2} \|\mathbf{e}(n)\|_2^2$$

$$= \frac{1}{2} \sum_d e_d^2(n)$$

$$= \frac{1}{2} \sum_d \left(\sum_{k \in \mathcal{S}_j} w^s_{j,d,k} h^s_{j,k}(n) - r_{j,d}(n) \right)^2. \tag{4.18}$$

In Eqn. (4.18), consider any term d of the outer summation. The inner summation is carried out over all columns $k \in \mathcal{S}_j$ of $\mathbf{W}^s_j$ such that $w^s_{j,k,d} \neq 0$. Noting that any such non-zero $w^s_{j,k,d} = p_j$, the error $\varphi(n)$ can be re-expressed in the following manner,

$$\varphi(n) = \frac{1}{2} p_j^2 \sum_{d \in \mathcal{D}_k} \left(\sum_{k \in \mathcal{S}_j} h^s_{j,k}(n) - \frac{r_{j,d}(n)}{p_j} \right)^2 + \frac{1}{2} \sum_{d \notin \mathcal{D}_k} r^2_{j,d}(n). \tag{4.19}$$

The summation in Eqn. (4.19) above is carried out over all elements in the set of indices $\mathcal{D}_k$ defined as, $\mathcal{D}_k = \{d | w^s_{j,d,k} \neq 0\}$. The set $\mathcal{D}_k$ can be obtained easily by examining k^{th} column of $\mathbf{W}^s_j$ and including all indices d, that have nonzero entries in that column. After scaling appropriately and ignoring the term in Eqn. (4.19) not containing $h^s_{j,k}(n)$, the objective is,

$$\varphi'(n) = \sum_{d \in \mathcal{D}_k} \left(\sum_{k \in \mathcal{S}_j} h^s_{j,k}(n) - r'_{j,d}(n) \right)^2 . \tag{4.20}$$

The proposed algorithm begins with $\mathbf{h}^s_j(n) = \mathbf{0}_{|\mathcal{S}_j|}$, updating it in a stepwise manner, one element in each step. In each step l a new index $k \in \mathcal{S}_j$ is selected and the corresponding $h^s_{j,k}(n)$ updated to 1. The algorithm can be implemented using a separate binary heuristic subroutine. However, the correct arguments need to be passed to the subroutine. The steps to do so are shown below.

$$\mathbf{r}_j(n) \leftarrow \mathbf{x}(n) - \mathbf{W}^f \mathbf{h}^f(n) - \sum_{j' \neq j} \mathbf{W}^s_{j'} \mathbf{h}^s_{j'}(n)$$

$$\mathbf{r}_j(n) \leftarrow \frac{1}{p_j} \mathbf{r}_j(n)$$

$$\mathbf{h}^s_j(n) \leftarrow \texttt{hillClimb}\left(\mathbf{W}^s_j, \mathbf{r}_j(n)\right)$$

The arguments of subroutine **`hillClimb()`** are a $D \times 1$ vector $\mathbf{r}$ and either a $D \times S$ binary matrix $\mathbf{W}$, or equivalently the sets of indices $\mathcal{D}_k$ for each $k \in \{1,2,\dots,S\}$. The subroutine returns an updated $\mathbf{h}$ that minimizes,

$$\varphi(\mathbf{h}|\mathbf{W}, \mathbf{r}) = \sum_d \left(r_d - \sum_{k=1}^{S} w_{k,d} h_k \right)^2 . \tag{4.21}$$

Alternately, the expression in Eqn. (4.21) can be written as,

$$\varphi(\mathbf{h}|\mathcal{D}_1, \dots, \mathcal{D}_D, \mathbf{r}) = \sum_{d \in \mathcal{D}_k} \left(r_d - \sum_{k=1}^{S} h_k \right)^2 + \frac{1}{2} \sum_{d \notin \mathcal{D}_k} r_d^2 , \tag{4.22}$$

The proposed subroutine is outlined in Algorithm 4.1. The two arguments that are passed to the subroutine **`hillClimb()`** consist of a basis matrix $\mathbf{W}$, and a residual vector $\mathbf{r}$. If $\mathbf{W} \equiv \mathbf{W}^s_j$, the basis matrix of the j^{th} shiftable load, then $\mathbf{r} \equiv \mathbf{r}_j(n)$ as shown in Eqn. (4.17) but after

normalization so that $p_j = 1$. It is assumed that two constants, S, the number of columns in $\mathbf{W}$, and L, the maximum allowable value of $\|\mathbf{h}\|_0$ are implicitly accessible to the subroutine.

Algorithm 4.1: Hillclimb Algorithm

`hillClimb`$(\mathbf{W}, \mathbf{r})$

1. $\mathbf{h} \leftarrow \mathbf{0}_S$
2. $\mathcal{L} \leftarrow \{1,2,\ldots,S\}$
3. $l \leftarrow 0$
4. terminate $\leftarrow$ **FALSE**
5. **for** each $k \in \mathcal{L}$
   - 5a. $\mathcal{D}_k \leftarrow \{d | w_{k,d} \neq 0\}$

   end
6. **while** terminate == FALSE
   - 6a. $\varphi_0 \leftarrow \sum_d r_d^2$
   - 6b. **for** each $k \in \mathcal{L}$
     - 6c. $\varphi_1(k) \leftarrow \sum_{\cdot} r_d^2 + \sum_{\cdot} (r_d - 1)^2$

     end
   - 6d. **if** $\min_k \varphi_1(k) \geq \varphi_0$ or $l > L$
     - 6e. terminate $\leftarrow$ TRUE
   - 6f. **else**
     - 6g. $k \leftarrow \operatorname{argmin}_k \varphi_1(k)$
     - 6h. $h_k \leftarrow 1$
     - 6i. $\mathcal{L} \leftarrow \mathcal{L} \backslash \{k\}$
     - 6j. $l \leftarrow l + 1$
     - 6k. **for** each $d \in \mathcal{D}_k$
       - 6l. $r_d \leftarrow r_d - 1$

       end

     end

   end

end

The subroutine **`hillClimb()`** maintains a set $\mathcal{L}$ of all indices k such that $h_k = 0$. Since the subroutine begins with $\mathbf{h} = \mathbf{0}_S$ (step 1), the set $\mathcal{L}$ is initialized to include all basis vectors in $\mathbf{W}$ (step 2). The quantity l stores the value of $\|\mathbf{h}\|_0$; therefore it is initialized to 1 (step 3). The subroutine maintains a Boolean variable `terminate` to indicate if the termination condition of the algorithm is satisfied; it is initialized to **`FALSE`** (step 4). Since $\mathbf{W}$ is a sparse binary matrix, the sets $\mathcal{D}_k$ ($k = 1,2, \dots, S$) are initialized to indicate the elements in column $\mathbf{w}_k$ in $\mathbf{W}$ that contain 1s (step 5).

During each iteration of the **`while`** loop (step 6), an h_k is updated to unity and the corresponding index k removed from $\mathcal{L}$. In other words, the norm $\|\mathbf{h}\|_0$ is increased by unity per iteration. As a hill-climbing procedure, in each iteration the subroutine picks an index k from $\mathcal{L}$ that lowers the error $\|\mathbf{r} - \mathbf{W}\mathbf{h}\|_2^2$ by the maximum amount. In step 6a, the current error is obtained and stored in the variable φ_0. In the **`for`** loop that follows in step 5, for every k in $\mathcal{L}$, the error $\varphi_1(k)$ that would result if the corresponding h_k were to be incremented to unity, is computed. However, before updating the h_k, the termination condition is evaluated. The indicator variable `terminate` is set to **`TRUE`** if $\|\mathbf{h}\|_0$ is equal to L, in which case no further updates are possible. The variable becomes **`TRUE`** also if incrementing any other h_k will only increase the error, that is, if the smallest entry of the vector $\boldsymbol{\varphi}_1$ exceeds φ_0. The latter situation arises when all elements of the residual $\mathbf{r}$ are less than 0.5. This is shown in step 6e.

If the termination condition is not satisfied, the subroutine proceeds by obtaining the index k that corresponds to the smallest $\varphi_1(k)$ (step 6f) and sets that h_k to 1 (step 6g). Next, the same index k is removed from the set $\mathcal{L}$, in step 6f. The variable l in incremented to indicate the new value of $\|\mathbf{h}\|_0$.

It is clear that the subroutine is a hill-climbing heuristic approach to update the weights of the shiftable load basis vectors. Therefore, upon termination, **`hillClimb()`** returns the binary vector $\mathbf{h} \equiv \mathbf{h}_j^S(n)$ such that,

$$\mathbf{h} = \underset{\|\mathbf{h}\|_0 \leq L}{\operatorname{arginf}} \|\mathbf{r} - \mathbf{W}\mathbf{h}\|_2^2 . \tag{4.23}$$

To the best of the my knowledge, there are no established upper bound estimates on the computational complexity of mean field annealing, as has been used in [48]. Although it is reasonable to consider $O\left(\left|\mathcal{S}_j\right|\right)$ as the per iteration complexity in extracting the largest singular

value of a sparse matrix (c.f. [91]), this is only an estimate; SVD algorithms as used in [92] for NMF, do not have strict upper bounds. In contrast, the hill climbing procedure described in this research requires $O(L_0 \ln|\mathcal{D}|)$ steps per iteration and L_0 can never exceed the total number of basis vectors $|\mathcal{S}_j|$.

4.3.4 NMF Algorithm

The proposed NMF algorithm follows a block coordinated descent method, with the matrices $\mathbf{W}^f$, $\mathbf{H}^f$, and each $\mathbf{H}_j^s$. Whereas the fixed load matrices are treated as in [88], [89], [90], updating the weight matrices of the shiftable loads, a novel feature of this research, is done by calling the subroutine **`hillClimb()`** described above multiple times in an iterative fashion. The overall algorithm is outlined in Algorithm 4.2.

Algorithm 4.2: NMF Algorithm

while converged == FALSE do

1. $\mathbf{W}^f \leftarrow \mathbf{W}^f \circ \frac{\left(\mathbf{X} \circ \widetilde{\mathbf{X}}^{\circ -1}\right)\mathbf{H}^{f\,\mathrm{T}}}{\mathbf{1}_{D\times N}\mathbf{H}^{f\,\mathrm{T}}}$
2. $\mathbf{W}^f \leftarrow \frac{\mathbf{W}^f}{\|\mathbf{W}^f\|}$
3. $\mathbf{H}^f \leftarrow \mathbf{H}^f \circ \frac{\mathbf{W}^{f\,\mathrm{T}}\left(\mathbf{X} \circ \widetilde{\mathbf{X}}^{\circ -1}\right)}{\mathbf{W}^{f\,\mathrm{T}}\mathbf{1}_{D\times N}}$
4. **for** each $n \in \mathcal{N}$
    5. **for** each j (in any order)
        5a. $\mathbf{r}_j(n) \leftarrow \mathbf{x}(n) - \mathbf{W}^f\mathbf{h}^f(n) - \sum_{j'\neq j}\mathbf{W}_{j'}^s\mathbf{h}_{j'}^s(n)$
        5b. $\mathbf{r}_j(n) \leftarrow \frac{1}{p_j}\mathbf{r}_j(n)$
        5c. $\mathbf{h}_j^s(n) \leftarrow$ **`hillClimb`** $(\mathbf{W}_j^s, \mathbf{r}_j(n))$
    end
end
end

For simplicity, the initialization steps are omitted. At first, the matrices $\mathbf{W}^f$, $\mathbf{H}^f$ are updated in accordance with Eqn. (4.13) and Eqn. (4.14). The updates are carried out in steps 1 and 3, with step 2 being the implementation of the normalization step in Eqn. (415). It may be noted that the informal notation used in step 2 should, in reality, be implemented iteratively to normalize each column of $\mathbf{W}^f$ separately.

In each iteration of step 4, a sample $n \in \mathcal{N}$ is picked, either randomly or sequentially, so that all shiftable load weights can be updated using the hill-climbing procedure. The appearance of a second level for loop in step 5 is due to the presence of more than a single shiftable load. As before, the inner loop may proceed in a sequential manner, or in any random order of the shiftable loads.

Given sample n and a shiftable load j, the residual vector is extracted in the manner shown in Eqn. (4.17). This is implemented in step 5a. The residual is normalized in step 5b, with respect to its peak p_j. Finally in step 5c, the column $\mathbf{h}_j^s(n)$ of the weight matrix $\mathbf{H}_j^s$ is updated by calling `hillClimb()`.

4.4 Simulation Results

The proposed approach was tested on actual energy usage profiles of a single residential customer that was obtained from the Pecan Street Inc., Dataport database [93] sampled at one minute intervals, and for the first three weeks in April, 2019. As energy consumption patterns on weekdays differ significantly from those in weekends, the latter was discarded, yielding a total of $N = 15$ samples. The data was arranged as a 1440×15 input matrix $\mathbf{X}$ whose columns were the 1440-dimensional sample vectors $\mathbf{x}(n), n \in \mathcal{N}$.

The database in [93] also included individual energy usages of the following four appliances, (*i*) a furnace, (*ii*) a washer/dryer unit, (*iii*) an oven, and (*iv*) kitchen appliances. These measurements provided the basis to evaluate the quality of the disaggregation obtained by the proposed algorithm. All four appliances were classified as shiftable loads for this purpose and indexed in the above order. The rest of the aggregate load was treated as fixed loads.

The peak values p_j of the duty cycle of each load $j \in \{1, 2, 3, 4\}$, as well as the maximum number of its ON cycles, L_j, of a typical day (denoted as $L_j^{\max}$) was determined through visual inspection. These parameters are shown in Table 4.1.

Table 4.1 Duty Cycle Parameters

	Furnace	Washer/Dryer	Oven	Kitchen appliances
$\boldsymbol{p_j}$	0.46	2.50	5.00	0.37
$\boldsymbol{L_j^{\max}}$	150.00	20.00	10.00	60.00

The observed L_j maximums (denoted as $L_j^{\max}$) of the loads in Table I served as upper limits on the number of ON cycles, so that constraints of the form $\left\|\mathbf{h}_j^s(n)\right\|_0 \leq L_j^{\max}$ were applied to each shiftable load. The total number of basis vectors were chosen to be $|\mathcal{F}| = 1$ for the fixed loads, and $\left|\mathcal{S}_j\right| = 1440$ for each shiftable load j. The disaggregated outcomes of the days indexed $n = 1$ and $n = 5$ (indexed after dropping weekends) were picked for illustrative purposes. These will be referred to hereafter as Day-1 and Day-2.

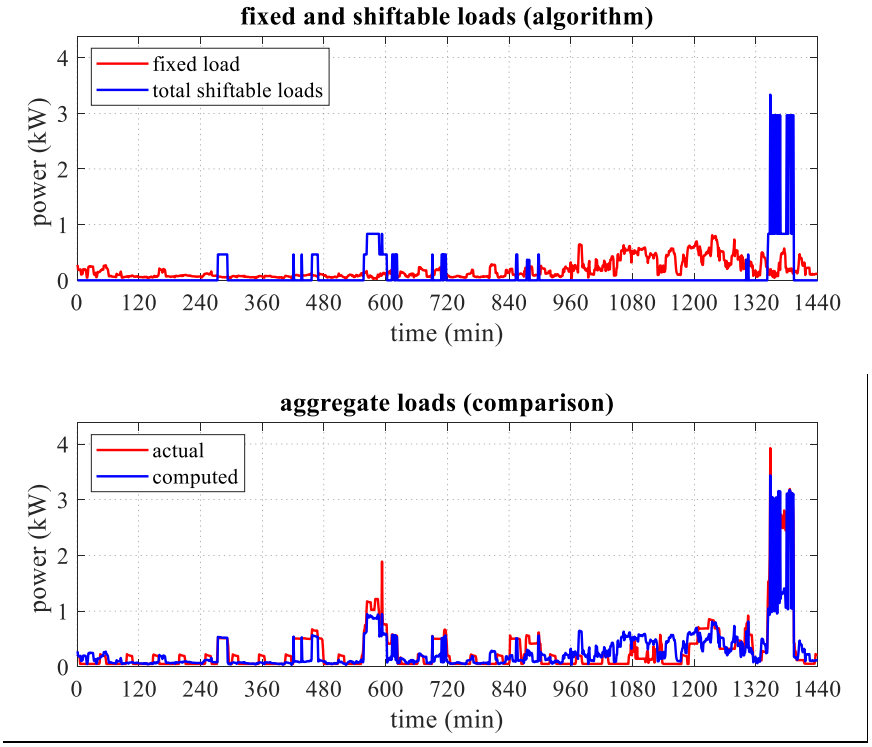

Figure 4.3: Disaggregated shiftable loads (top) and aggregate loads (bottom) for Day-1.

The plots in Figure 4.4 and Figure 4.5 show the results obtained by the NMF algorithm for Day-1. The disaggregated fixed load $\mathbf{W}^f\mathbf{h}^f(1)$ (red) and the sum of all four shiftable loads, $\sum \mathbf{W}_j^s\mathbf{h}_j^s(1)$ (blue) for the entire 24 hour period appears in Figure 4.4. (top). For the sake of comparison, the expected aggregate load $\tilde{\mathbf{x}}(1)$ from the algorithm (blue) is plotted alongside the real load profile $\mathbf{x}(1)$ (red) in Figure 4.4 (bottom). The effectiveness of the proposed approach is evident from the similar patterns of both plots. The peak consumption in both cases occur in the late evening hours (1320 – 1440 mins). Additionally the real data shows increased energy usage in the morning hours (540 – 600 mins), which is effectively reproduced by the NMF algorithm.

The disaggregated loads of the individual appliances for Day-1 are provided in Figure 4.5. The plots for the furnace, washer/dryer unit, oven, and kitchen appliances appear in order (top through bottom). The actual load data of this 24 hour period appears in red.

Upon close observation, it is apparent that the NMF algorithm accurately reproduces the ON and OFF periods for each load (blue). During Day-1, there were no ON periods for the oven in either case. The ON periods for the washer/dryer unit occur in the interval 1340 – 1400 mins. The effectiveness of the NMF algorithm is obvious from the very strong resemblance of the disaggregated washer/dryer usage to the real data. Likewise, the NMF algorithm faithfully reproduces the real furnace usage profile, albeit to a somewhat lesser extent than before. In comparison to the others, there are some discrepancies in the usage profile that the NMF algorithm yields and the real data for the kitchen appliances.

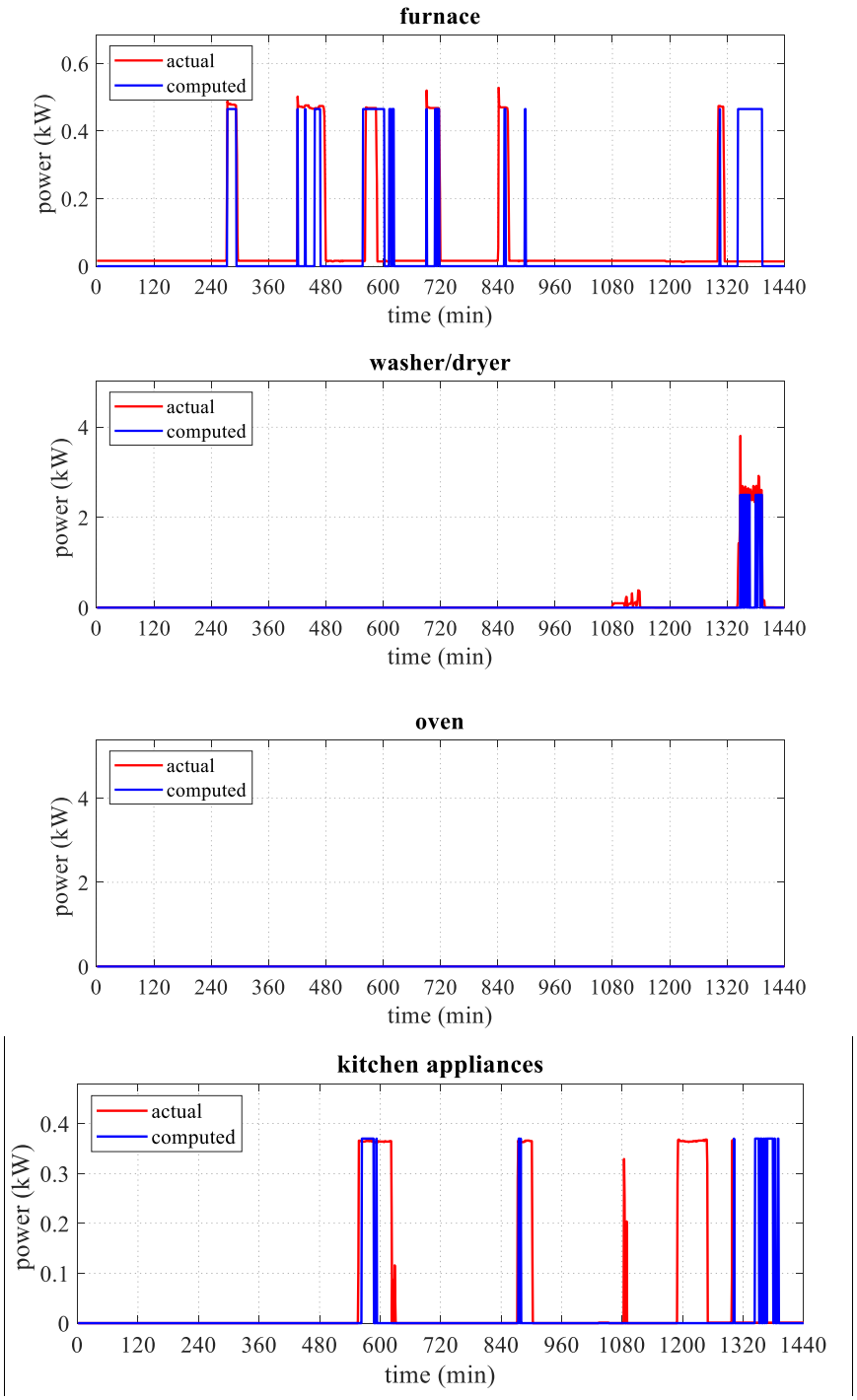

Figure 4.4: Disaggregated shiftable loads for Day-1.

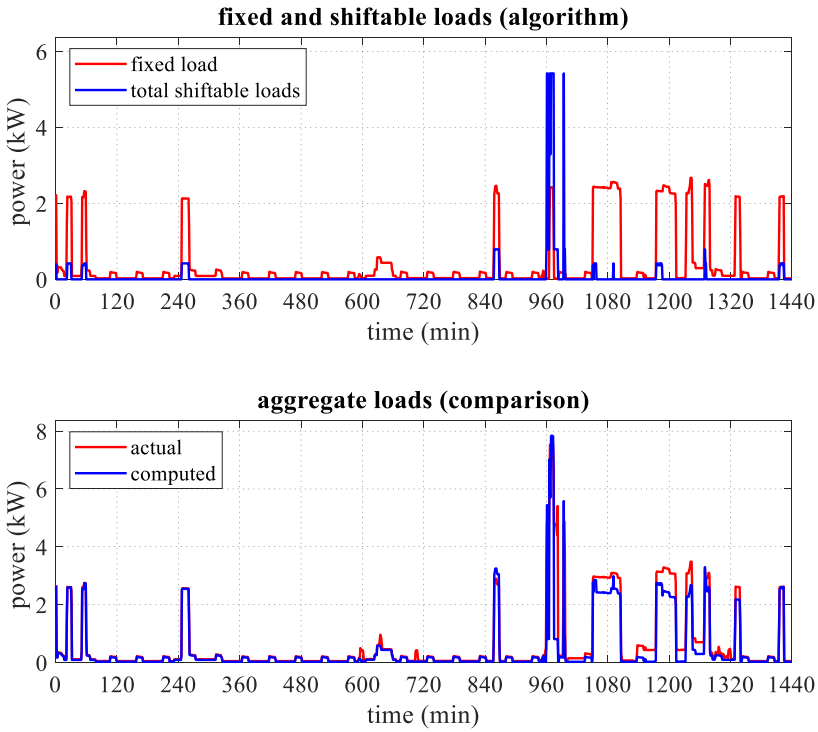

Figure 4.5: Disaggregated shiftable loads (top) and aggregate loads (bottom) for Day-2.

Figure 4.6 and Figure 4.7 pertain to the energy usage occurring in Day-2. As the results are remarkably similar to those in Day-1, we focus only on a few key observations. In contrast to the previous results, from the actual data in Figure 4.6 (bottom) it is seen that the increased usage of shiftable loads during the evening is more spread out. This feature is reflected in the disaggregated signals. Unlike before, there are a few ON cycles for the oven, which is again captured by the algorithm. As in the previous case, it can be seen that the algorithm is not able to pick the ON cycles of the kitchen appliance with the same precision as with other shiftable loads. We attribute the difference to the smaller peak value $p_4 = 0.37$ relative to those of the other appliances.

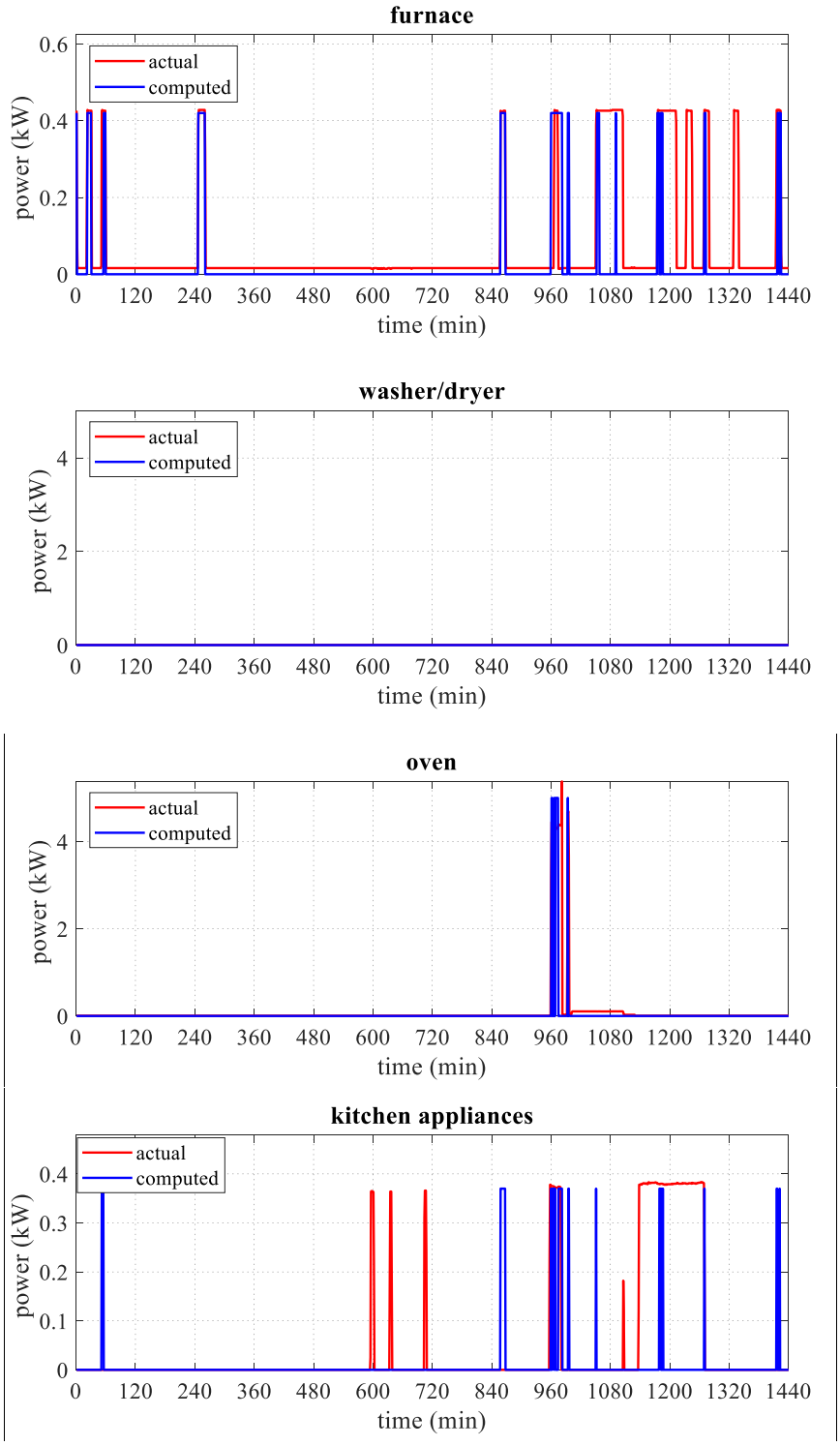

Figure 4.6: Disaggregated shiftable loads for Day-2.

4.5 Conclusion

The major contributions of this study are as follows.

(*i*) The load disaggregation of real input data at a significantly high resolution of 1 minute intervals clearly demonstrates the efficacy of the NMF algorithm suggested in this research.

(*ii*) Categorizing individual appliances into fixed and shiftable loads allowed the proposed NMF based algorithm to leverage the observed characteristics of each appliance's ON-cycle.

(*iii*) It has been shown that the NMF algorithm, including the treatment of the shiftable loads, can be viewed in terms of maximum likelihood. This provides a theoretical justification of the new approach introduced in this research.

(*iv*) The hill climbing heuristic, requiring up to only L_0 steps per iteration of the outer loop, offers a significant computational advantage over other NMF approaches if used for a similar application. (As an example, this algorithm requires up to $L_0 = L_1^{\max} = 150$ steps for the furnace, whereas an SVD based NMF [92], would need $|\mathcal{S}_1| = 1440$ steps).

In spite of the high fidelity of the disaggregated loads obtained using the approach introduced in this research, there is ample scope for future research. Incorporating a semi-supervised learning algorithm to obtain best fits of the duty cycles would obviate the need of prior knowledge of the duty cycles of individual appliances. Another potential enhancement is to allow the algorithm to learn the ON cycles' probabilities of the appliances. The algorithm may be extended to include these probabilities within the theoretical framework.

Chapter 5 - A Data Driven Approach for Energy Consumer Modeling

The uses of reliable estimates of instantaneous consumer load demands *a priori* are manifold. They are routinely in short-term planning, load scheduling, security and privacy issues, energy trading, and cooperative energy use. Non-parametric models, such as neural networks, are sufficient in some of these applications which only require load forecasts. Elsewhere, separate estimates of fixed and shiftable loads are required. A semi-parametric estimation model is needed in market-oriented aspects of the grid, costs sensitivities of demands must be known. In the latter case, research work consistently uses somewhat arbitrary parameters that seem to work best. We propose a generic class of semiparametric models that is derived from real consumer data. A two-step machine learning approach is used. In the first phase, disaggregation of the load into fixed and shiftable components is accomplished by means of a hybrid non-negative matrix factorization (NMF) and Gaussian mixture models (GMM) approach, the latter trained with the expectation-maximization (EM) algorithm. The fixed and shiftable loads are subject to analytic treatment using economic considerations. Lastly, the model parameters are estimated using an $\boldsymbol{L_2}$-norm support vector regression (SVR).

5.1 Introduction

Aggregate data refers to higher level data that is composed of multiple components. For example, traffic flow of a highway from a large city comprises of flows from all arterials leading to the highway. Likewise, the daily oil production of a country is the aggregate of the that produced by all oil corporations that are located in it. In the latter example, oil production can be divided into one component that directly responsive to supply, while another component that remains fixed, due to physical drilling requirements of oil wells. For capacity planning, environmental regulations, economic policies, and several other purposes, a significant amount of information can be acquired by disaggregating the total oil production into these two components alone, without having to look into those of individual oil companies.

This research envisages an analogous situation in the consumption of energy by households within the energy distribution grid. There are a variety of loads present in a typical household, including essentials such as lighting or temperature control appliances, whose use being entirely dependent upon the time of day and the surrounding temperatures is not influenced by economic considerations. The household also contains other loads such as PHEV charging that are directly driven by time of use (TOU) pricing. Accordingly, energy consumption can be divided into two categories, (*i*) *fixed loads*, and (*ii*) *shiftable loads*. Further details of this dichotomy are postponed to a later section. Unfortunately, in contrast to other domains where individual components of the aggregate data are readily available, however any residential consumer uses energy is treated as private, inaccessible information. In other words, decomposing total usage patterns can only be estimated using computational methods. This task is referred to as load disaggregation.

There are significant implications from load disaggregation in energy trading – the procurement of energy by a customer from another customer, or an upper level grid entity such as aggregator or DSO, in exchange for money, through auction mechanisms that may lead to Nash, Stackelberg, or other game theoretic equilibrium conditions. Each customer simply tries to maximize its own payoff, i.e. the difference between the *utility* gained from consuming a certain amount of energy and the price it pays to procure that amount. The utility in this context quantifies the amount of satisfaction that a consumer derives from using a certain amount of energy. In accordance with the classic econometric precept – the law of diminishing returns, the utility function should ideally be monotonically non-decreasing and quasi-concave. For mathematical tractability, it must also be Lipschitz continuous. The derivative of any utility function that is differentiable is its *marginal utility*. In auctions, customer agents place cost bids that are equal to marginal utilities.

Disaggregated loads are used in this research to obtain a nonlinear utility model of the residence. The model is semi-parametric as it does not require fine-grained disaggregation of the smart meter data into individual appliances; disaggregation of a coarser granularity to divide the aggregate into fixed and shiftable components are sufficient for the semi-parametric consumer model proposed here. In this context, parametric models are ones that require adequate representations of all individual appliances. At the other end of the spectrum are non-parametric models that are entirely empirical.

5.2 Overall Approach

It is assumed that any user's energy usage x_t at any time instant t is divided into two components, i.e.

$$x_t = x_t^f + x_t^s + \eta_t. \tag{5.1}$$

Here x_t^f is the fixed load and x_t^s, the shiftable load. The quantity η_t is the sensor error.

We define the following vectors, $\mathbf{x} = [x_t]_{t\in\mathcal{T}}$, $\mathbf{x}^f = \left[x_t^f\right]_{t\in\mathcal{T}}$, and $\mathbf{x}^s = [x_t^s]_{t\in\mathcal{T}}$. Each quantity is a $|\mathcal{T}| \times 1$ vector, where $\mathcal{T}$ is the set of time instants.

The time instances $t \in \mathcal{T}$ are divided into periods with each period including exactly L time instances. The set of periods is $\mathcal{K}$ so that $L = |\mathcal{K}|^{-1}|\mathcal{T}|$. For example, if the time instances are of a minute duration each, that are divided up into hour-long periods, then $|\mathcal{T}| = 1440$ mins/day, $|\mathcal{K}| = 24$ hours/day, and $L = 60$ mins/hour. Within each period $k \in \mathcal{K}$ we define the $L \times 1$ vectors $\mathbf{x}_k^f = \left[x_t^f\right]_{\kappa(t)=k}$, $\mathbf{x}_k^s = [x_t^s]_{\kappa(t)=k}$, where $\kappa(\cdot)$ yields the period index corresponding to a time instance t. The temperature vector $\boldsymbol{\theta} = [\theta_k]_{k\in\mathcal{K}}$ and the vector of unit costs $\mathbf{c} = [c_k]_{k\in\mathcal{K}}$ are other $L \times 1$ vectors.

The dataset used in this research consists of multiple samples, where each sample pertains to a day. Accordingly, we define the matrices, $\mathbf{X}^f = [\mathbf{x}^f(n)]_{n\in\mathcal{N}}$, and $\mathbf{X}^s = [\mathbf{x}^s(n)]_{n\in\mathcal{N}}$, where $\mathcal{N}$ is the set of samples and sample index n indicates a day. With vec$\cdot$ being the vector operator, vec $\mathbf{X}^s$ and vec $\mathbf{X}^f$ define two $|\mathcal{T}||\mathcal{N}| \times 1$ vectors of loads. The $L \times |\mathcal{K}|$ matrix of temperatures is $\boldsymbol{\Theta} = [\boldsymbol{\theta}(n)]_{n\in\mathcal{N}}$; vec $\boldsymbol{\Theta}$ is its equivalent $|\mathcal{K}||\mathcal{N}| \times 1$ vector. The cost vector $\mathbf{c}$ is constant across all samples in $\mathcal{N}$.

5.2.1 Load Model

Fixed loads pertain to appliances that are deemed essential in any residence, such as lighting, refrigeration, and heating equipment. Consequently unit energy price has no bearing on their operation. Most fixed loads cannot be characterized as discrete loads. For instance illumination and heating are needed in each room and would require too many states in an HMM. Their energy consumption is usually relatively low, including refrigerators that do display ON-OFF cycles. Lastly, these appliances are in operation throughout any 24 hour period. Accordingly, it is assumed in this research that fixed loads are, (*i*) typically continuous loads, (*ii*) independent of unit price,

(*iii*) used throughout the day, (*iv*) relatively low valued. Figure 5.1 (top) illustrates two fixed loads (brown, magenta).

The schematic in Figure 5.2 (bottom) shows three shiftable appliances (blue, green, cyan) drawing a fixed amount of power when activated. Occasionally, an appliance may display multiple peaks (cyan). A composite peak appears when two or more appliances are simultaneously active. In the schematic, two such loads (blue and green) are active in the same instant).

5.2.1.1 Fixed Load

It is assumed that the fixed load at any period k consists of a temperature dependent term so that,

$$x_k^f = p_k + q_k\theta_k. \tag{5.2}$$

Here p_k and q_k are two model parameters, $x_k^f = \mathbf{1}_L^{\mathrm{T}}\mathbf{x}_k^f$ is the total fixed load and θ_k is the temperature during that period.

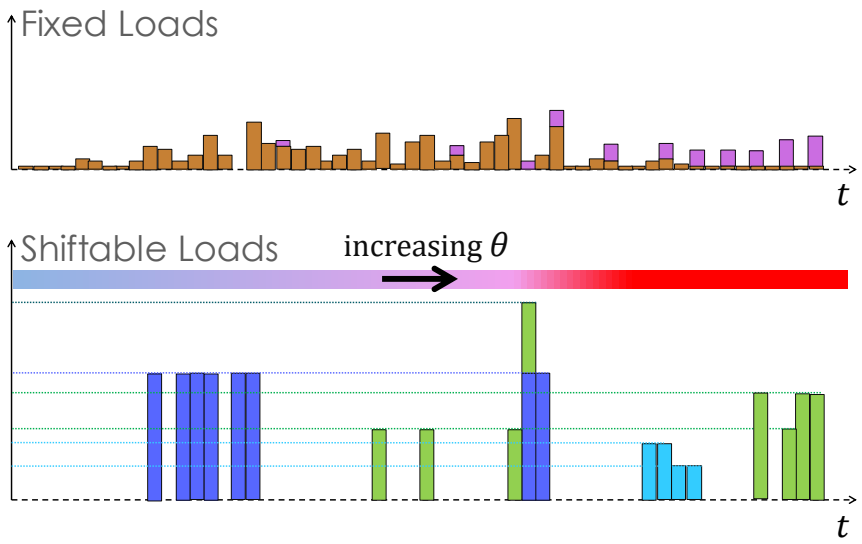

Figure 5.1: Schematic showing typical profiles of fixed (top) and shiftable (bottom) loads.

5.2.1.2 Shiftable Loads

Consider a market with two divisible resources, indexed 1 and 2. Suppose the amounts consumed by any agent are x_1 and x_2. The agent's preference can be quantified in terms of the non-decreasing function, $y(x_1, x_2)$. When the resources display perfect complementarity for the agent, the latter's preference can be expressed as $y(x_1, x_2) = \min(a_1x_1, a_2x_2)$. Here a_1 and a_2 are two constants that are based on the agent's individual characteristics. On the other hand, when the

commodities are regarded as perfect substitutes of each other, then $y(x_1, x_2) = a_1x_1 + a_2x_2$. The Cobb-Douglas function [94] is used as the preference function when the agent treats the two resources as intermediate between being perfect complements and perfect substitutes (see Figure 5.2.). In the present case with two resources, the preference can be expressed as $y(x_1, x_2) = x_1^{a_1}x_2^{a_2}$. In general with **x** representing the vector of all resources, an agent's Cobb-Douglas preference is $y(\mathbf{x}) = \prod_k (x_k)^{a_k}$. The parameters $a_k > 0$ are referred to as elasticity constants. Some shiftable loads can display perfect complementarity, whereas others, perfect substitution. Therefore, the Cobb-Douglas preference is widely considered to be a suitable means to model a consumer's preference.

However, a few enhancements are required to make the preference amenable for our purpose. First, a scaling factor is incorporated to reflect individual user's characteristics. For instance, a household may require more energy to derive the same benefit of consuming energy as either a more frugal or an environmentally aware one. Next, since the original Cobb-Douglas preference function serves as a production function, it becomes zero if any component $x_k = 0$. Therefore each factor is incremented by 1 to prevent this from happening, so that $y(\mathbf{x}^s) = \prod_{k\in\mathcal{K}} (b^{-1}x_k^s + 1)^{a_k}$. The logarithm of the modified Cobb–Douglas preference is chosen as the utility function. Treating the logarithm of the Cobb-Douglas preference in this manner, introduces several desirable features to the utility function: (*i*) it is strictly concave and increasing, (*ii*) it is Lipschitz continuous and differentiable, and (*iii*) it conveniently intersects the origin at $\mathbf{x}^s = \mathbf{0}$, indicating that a consumer can glean no utility without consuming any energy. The expression for the utility used here is given by,

$$u(\mathbf{x}^s) = \sum_{k\in\mathcal{K}} a_k \log\left(b^{-1}\left(x_k^s - (x_{k,0}^s + d_k\theta_k)\right)_+ + 1\right). \tag{5.3}$$

The derivative of $u(\mathbf{x}^s)$ with respect to any x_k^s is the agent' marginal utility of that period k, which is also equal to the unit cost.

$$c_k = \frac{a_k}{x_k^s - \left(x_{k,0}^s + d_k\theta_k\right) + b}. \tag{5.4}$$

It can be seen in Eqn. (5.3) and Eqn. (5.4) that an amount of energy $(x_{k,0}^s + d_k\theta_k)$ is subtracted from x_k^s. The quantity $x_{k,0}^s$ is the base shiftable load. It is the component of x_k^s that is determined by non-economic factors (e.g. it is not practicable for a washer/dryer unit to be put into

use at 2:00 AM). The term $d_k\theta_k$ is the temperature dependent component of x_k^s (such as load due to air conditioning). Generally speaking, the energy consumption increases when the ambient temperature deviates from some desirable value, say θ^{des}. Under these circumstances a term $d\left|\theta_k - \theta^{\text{des}}\right|$ should have been added to x_k^s. However, as the specific data used in this research involved only warmer days, it is assumed that $\theta_k > \theta^{\text{des}}$, thereby justifying the inclusion of an additive term $d_k\theta_k$ in Eqn. (5.3) and Eqn (5.4). From Eqn. (5.4), the total shiftable load is,

$$x_k^s = c_k^{-1}a_k - b + x_{k,0}^s + d_k\theta_k. \tag{5.5}$$

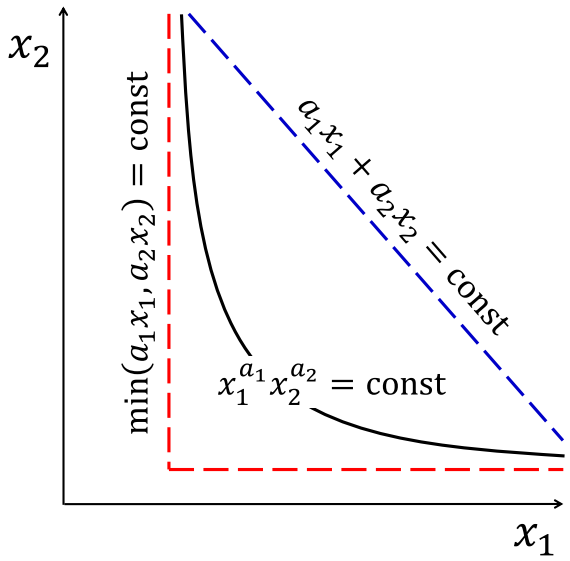

Figure 5.2: Isoquant for a Cobb-Douglas preference function (black). For comparison, isoquants for perfect substitutes (blue) and perfect complements (red) are also shown in dashed lines.

5.2.3 Algorithmic Framework

The load matrix **X**, the temperature matrix **Θ**, and the vector **c** of unit costs are the three input data sets used in this research. Load disaggregation is carried out using the hybrid approach proposed here, with the GMM associated with shiftable loads and NMF with fixed loads. The hybrid NMF-GMM disaggregation algorithm produces matrices of fixed loads $\widetilde{\mathbf{X}}^f$, and shiftable loads, $\widetilde{\mathbf{X}}^s$, which serve as the input to the parameter estimation algorithm that yields all parameters associated with the consumer utility model. Figure 5.3 illustrates the various steps of this approach.

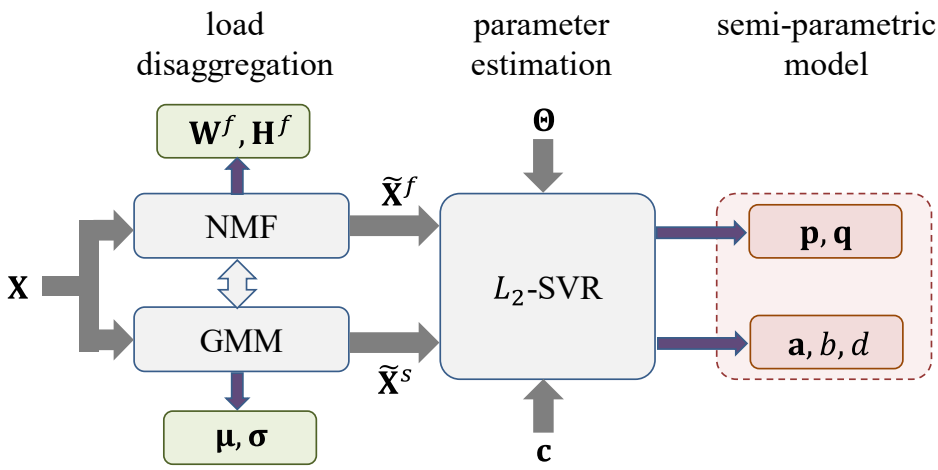

Figure 5.3: Steps used in the proposed approach. Arrows depict flow of information

5.3 Load Disaggregation

5.3.1 Gaussian Mixture Model

The shiftable loads are represented in our framework in terms of a Gaussian mixture model (GMM). The j^{th} prior, mean, and variance are π_j μ_j, and σ_j^2, which are trainable parameters with the exception of μ_0 which is permanently assigned a value $\mu_0 = 0$ to represent the case when none of the shiftable loads is in use. Its variance σ_0^2 is treated as a nonzero to subsume noise. With G being the number of nonzero Gaussians, the probability distribution of x_t can be expressed as per the following expression,

$$pr[x_t] = \sum_{j=0}^{G} \pi_j e^{-\frac{(x_t-\mu_j)^2}{\sigma_j^2}} . \tag{5.5}$$

In addition, the binary variable $\delta_{g(t,n)=j}$ is used to indicate that the j^{th} Gaussian is 'active' at time t, in sample n.

The GMM is trained using the expectation-maximization (EM) algorithm, briefly described here for convenience. The algorithm consists of an expectation step (E-step) where, using existing estimates of π_j μ_j, and σ_j^2 for each j, the posterior probabilities, $z_{j,t}(n) = pr[\delta_{g(t)=j}|x_t]$ of each Gaussian is estimated as,

$$\mathrm{z}_{j,t}(n) = \Sigma_{t,n}^{-1}\pi_j e^{-\frac{\left(x_t(n)-\mu_j\right)^2}{\sigma_j^2}}. \tag{5.6}$$

The quantity $\Sigma_{t,n}$ in the denominator is for normalization, such that the joint probability is unity, $\sum_{j=0}^{G}\mathrm{z}_{j,t}(n) = 1$, so that,

$$\Sigma_{t,n} = \sum_{j=0}^{G}\pi_j e^{-\frac{\left(x_t(n)-\mu_j\right)^2}{\sigma_j^2}}. \tag{5.7}$$

In in this manner, the posteriors for each time instance $t \in \mathcal{T}$ of each sample $n \in \mathcal{N}$ are obtained in the E-step.

During the maximization step, the GMM parameters, π_j μ_j, and σ_j^2 are updated using the following rules,

$$\mu_j = \sum_{t\in\mathcal{T},n\in\mathcal{N}}\mathrm{z}_{j,t}x_t(n), \tag{5.8}$$

$$\sigma_j^2 = \sum_{t\in\mathcal{T},n\in\mathcal{N}}\mathrm{z}_{j,t}\left(x_t(n)-\mu_j\right)^2, \tag{5.9}$$

$$\pi_j = |\mathcal{T}|^{-1}|\mathcal{N}|^{-1}\sum_{t\in\mathcal{T},n\in\mathcal{N}}\mathrm{z}_{j,t}(n). \tag{5.10}$$

The E-step and M-step are repeated multiple times until convergence.

Given a total load $x_t(n)$, the shiftable load is estimated according to the expression below,

$$\tilde{x}_t^s(n) = \max_j \mu_j \le x_t(n). \tag{5.11}$$

Ignoring noise, the remaining load $x_t(n) - x_t^s(n)$ is taken to be the fixed load $\tilde{x}_t^f(n)$, i.e.,

$$\widetilde{\mathbf{X}}^f = \mathbf{X} - \widetilde{\mathbf{X}}^s. \tag{5.12}$$

The estimated fixed load $\widetilde{\mathbf{X}}^f$ is used in the following algorithm.

5.3.2 Non-Negative Matrix Factorization

It is assumed that the shiftable load $\tilde{\mathbf{x}}^f(n)$ of each sample $n \in \mathcal{N}$ can be represented using a set of B basis vectors of dimensionality $|\mathcal{T}| \times 1$. If $h_k(n), k = 1, \dots, B$ are their coefficients, we must have,

$$\tilde{\mathbf{x}}^f(n) = \sum_{k=1}^{B} h_k^f(n)\mathbf{w}_k^f. \tag{5.13}$$

The basis set is assumed to be orthonormal, so that $\mathbf{w}_k^{\mathrm{T}}\mathbf{w}_l = \delta_{k=l}$. These basis vectors form the columns of a $|\mathcal{T}| \times B$ basis matrix, $\mathbf{W}^f = [h_k(n)]_{k\in\mathcal{K}}$. Similarly, letting $\mathbf{h}^f(n) = \left[h_k^f(n)\right]_{k\in\mathcal{K}}$ be the $B \times 1$ vector of coefficients in Eqn. (5.13), we define the $B \times |\mathcal{N}|$ matrix, $\mathbf{H}^f = [\mathbf{h}^f(n)]_{n\in\mathcal{N}}$. Therefore the above expression can be written more concisely as $\tilde{\mathbf{X}}^f = \mathbf{W}^f\mathbf{H}^f$. As the basis vectors are mutually orthonormal, it is apparent that ${\mathbf{W}^f}^{\mathrm{T}}\mathbf{W}^f = \mathbf{I}_{B\times B}$.

The matrices $\mathbf{W}^f$ and $\mathbf{H}^f$ are trained using the usual multiplicative update rule [38], [49], [50], [51] to minimize the squared Frobenius norm $\Phi = \left\|\mathbf{X}^f - \tilde{\mathbf{X}}^f\right\|_F^2$. Consider any parameter $\mathbf{P}$ (which can be either $\mathbf{W}^f$ or $\mathbf{H}^f$). The gradient, $\nabla_{\mathbf{P}}\Phi$ of Φ can be expressed in terms of its positive and negative components as, $\nabla_{\mathbf{P}}\Phi = \nabla_{\mathbf{P}}^{+} - \nabla_{\mathbf{P}}^{-}$. The update rule using the multiplicative method is $\mathbf{P} \leftarrow \mathbf{P} \circ \nabla_{\mathbf{P}}^{-} \oslash \nabla_{\mathbf{P}}^{+}$. The coefficient matrix $\mathbf{H}^f$ is updated in this manner,

$$\mathbf{H}^f \leftarrow \mathbf{H}^f \circ \frac{{\mathbf{W}^f}^{\mathrm{T}}\tilde{\mathbf{X}}^f}{{\mathbf{W}^f}^{\mathrm{T}}(\mathbf{W}^f\mathbf{H}^f)}. \tag{5.14}$$

The corresponding update rule for $\mathbf{W}^f$ can be obtained in the same manner. However, in order to maintain orthogonality of the basis vectors, a modified version has been adopted in this research [95]. Accordingly,

$$\mathbf{W}^f \leftarrow \mathbf{W}^f \circ \frac{\tilde{\mathbf{X}}^f{\mathbf{H}^f}^{\mathrm{T}}}{(\mathbf{W}^f\mathbf{H}^f){\tilde{\mathbf{X}}^f}^{\mathrm{T}}\mathbf{W}^f}. \tag{5.15}$$

5.4 Parameter Estimation

5.4.1 Constraints

The $|\mathcal{K}| \times 1$ vectors of parameters are, $\mathbf{a}$, $\mathbf{p}$, and $\mathbf{q}$. The parameters b and d, which are the scaling factors that are intrinsic to the household, are maintained at the same value everywhere. We define $\mathbf{G}$ to be a $|\mathcal{K}| \times |\mathcal{T}|$ aggregating matrix that sums time samples of variables index t to their totals, that are indexed k. Letting $\mathcal{k}(t)$ be the period of time instance t, $G_{k,t} = \delta_{k=\mathcal{k}(t)}$ where $\delta_\sigma \in \{0,1\}$

is the usual Kronecker delta that is 1 when the statement 'σ' appearing in the subscript is true, or 0 when 'σ' if false.

$$\mathbf{V} = \begin{bmatrix} \mathbf{1}_L^{\mathrm{T}} & \cdots & \mathbf{0}_L^{\mathrm{T}} \\ \vdots & \ddots & \vdots \\ \mathbf{0}_L^{\mathrm{T}} & \cdots & \mathbf{1}_L^{\mathrm{T}} \end{bmatrix}. \tag{5.16}$$

We introduce an error bound on the difference between parametric estimate of the money m_k^s from Eqn. (5.1) and the actual money expended $c_k x_k^s$. Let ϵ_1^f be the maximum allowable error on the fixed loads,

$$\left|\mathbf{p} + \boldsymbol{\theta}(n) \circ \mathbf{q} - \mathbf{V}\mathbf{x}^f(n)\right| \leq \mathbf{1}_{|\mathcal{K}|}\epsilon^f + \boldsymbol{\xi}^f. \tag{5.17}$$

In a similar manner, let ϵ_1^s be the allowable tolerance of the error in the estimated money due to the shiftable loads as in Eqn. (5.3), and the real amount from the data,

$$\left|\mathbf{c}^{\circ -1} \circ \mathbf{a} - \mathbf{1}_{|\mathcal{K}|}b + \boldsymbol{\theta}(n) \circ \mathbf{d} + \mathbf{x}_0^s - \mathbf{V}\mathbf{x}^s(n)\right| \leq \mathbf{1}_{|\mathcal{K}|}\epsilon^s + \boldsymbol{\xi}^s. \tag{5.18}$$

5.4. 2 Objective

The objective function contains for the error vectors $\boldsymbol{\xi}^f$ and $\boldsymbol{\xi}^s$.

$$\Omega = \frac{1}{2}\left(w^f \boldsymbol{\xi}^{f\mathrm{T}}\boldsymbol{\xi}^f + w^s \boldsymbol{\xi}^{s\mathrm{T}}\boldsymbol{\xi}^s + \gamma^a \mathbf{a}^{\mathrm{T}}\mathbf{a} + \gamma^b b^2 + \gamma^x \mathbf{x}_0^{s\mathrm{T}}\mathbf{x}_0^s + \gamma^d \mathbf{d}^{\mathrm{T}}\mathbf{d} + \gamma^p \mathbf{p}^{\mathrm{T}}\mathbf{p} + \gamma^q \mathbf{q}^{\mathrm{T}}\mathbf{q}\right). \tag{5.19}$$

5.4.3 Quadratic Programming

With multiple samples $n \in \mathcal{N}$ the expressions in Eqn. (5.17) and Eqn. (5.18) yield $2|\mathcal{K}||\mathcal{N}|$ constraints. The objective function can be reformulated as the sums of L_2 norms. These yield the following quadratic programming problem with linear equality constraints for parameter estimation.

Minimize,

$$\Omega = \frac{1}{2}\left(\left\| \begin{bmatrix} w^f \mathbf{I}_{|\mathcal{K}|\times|\mathcal{K}|} & \mathbf{0}_{|\mathcal{K}|\times|\mathcal{K}|} \\ \mathbf{0}_{|\mathcal{K}|\times|\mathcal{K}|} & w^s \mathbf{I}_{|\mathcal{K}|\times|\mathcal{K}|} \end{bmatrix}^{\frac{1}{2}} \begin{bmatrix} \boldsymbol{\xi}^f \\ \boldsymbol{\xi}^s \end{bmatrix} \right\|^2 \right.$$

$$+ \left\| \begin{bmatrix} \gamma^a \mathbf{I}_{|\mathcal{K}|\times|\mathcal{K}|} & \mathbf{0}_{|\mathcal{K}|\times 1} & \mathbf{0}_{|\mathcal{K}|\times|\mathcal{K}|} & \mathbf{0}_{|\mathcal{K}|\times|\mathcal{K}|} \\ \mathbf{0}_{1\times|\mathcal{K}|} & \gamma^b & \mathbf{0}_{1\times|\mathcal{K}|} & \mathbf{0}_{1\times|\mathcal{K}|} \\ \mathbf{0}_{|\mathcal{K}|\times|\mathcal{K}|} & \mathbf{0}_{|\mathcal{K}|\times 1} & \gamma^d \mathbf{I}_{|\mathcal{K}|\times|\mathcal{K}|} & \mathbf{0}_{|\mathcal{K}|\times|\mathcal{K}|} \\ \mathbf{0}_{|\mathcal{K}|\times|\mathcal{K}|} & \mathbf{0}_{|\mathcal{K}|\times 1} & \mathbf{0}_{|\mathcal{K}|\times|\mathcal{K}|} & \gamma^x \mathbf{I}_{|\mathcal{K}|\times|\mathcal{K}|} \end{bmatrix}^{\frac{1}{2}} \begin{bmatrix} \mathbf{a} \\ b \\ \mathbf{d} \\ \mathbf{x}_0^s \end{bmatrix} \right\|^2$$

$$\left. + \left\| \begin{bmatrix} \gamma^p & \mathbf{0}_{|\mathcal{K}|\times|\mathcal{K}|} \\ \mathbf{0}_{|\mathcal{K}|\times|\mathcal{K}|} & \gamma^q \end{bmatrix}^{\frac{1}{2}} \begin{bmatrix} \mathbf{p} \\ \mathbf{q} \end{bmatrix} \right\|^2 \right)$$

Subject to,

$$\begin{cases} -\boldsymbol{\xi}^f + \mathbf{p} + \mathrm{diag}(\boldsymbol{\theta}(n))\mathbf{q} \le \mathbf{1}_{|\mathcal{K}|}\epsilon^f + \mathbf{V}\mathbf{x}^f(n), \\ -\boldsymbol{\xi}^f - \mathbf{p} - \mathrm{diag}(\boldsymbol{\theta}(n))\mathbf{q} \le \mathbf{1}_{|\mathcal{K}|}\epsilon^f - \mathbf{V}\mathbf{x}^f(n); \end{cases} \quad \forall n \in \mathcal{N}.$$

$$\begin{cases} -\boldsymbol{\xi}^s + \mathrm{diag}(\mathbf{c}^{\circ -1})\mathbf{a} - \mathbf{1}_{|\mathcal{K}|}b + \mathrm{diag}\big(\boldsymbol{\theta}(n)\big)\mathbf{d} + \mathbf{x}_0^s \le \mathbf{1}_{|\mathcal{K}|}\epsilon_1^s + \mathbf{V}\mathbf{x}^s(n), \\ -\boldsymbol{\xi}^s - \mathrm{diag}(\mathbf{c}^{\circ -1})\mathbf{a} + \mathbf{1}_{|\mathcal{K}|}b - \mathrm{diag}\big(\boldsymbol{\theta}(n)\big)\mathbf{d} - \mathbf{x}_0^s \le \mathbf{1}_{|\mathcal{K}|}\epsilon_1^s - \mathbf{V}\mathbf{x}^s(n). \end{cases} \quad \forall n \in \mathcal{N}.$$

5.5 Simulation Results

5.5.1 Data Processing

The proposed approach was tested on actual energy usage data of two residential customers (User-1 and User-2) that was obtained from the Pecan Street Inc., Dataport database [93] sampled at one minute intervals, and for 61 days in March - April, 2018 so that $\mathcal{T} = \{1,2,\ldots 1440\}$, and $\mathcal{N} = \{1,2,\ldots 61\}$. The total energy data was arranged as a 1440×61 input matrix $\mathbf{X}$ whose columns were the 1440-dimensional sample vectors $\mathbf{x}(n), n \in \mathcal{N}$. The database also included individual appliance usage measurements at the circuit level. The following are the appliances in use, (*i*) air conditioner, (*ii*) electric car, (*iii*) washer/dryer, (*iv*) dishwasher, (*v*) microwave, (*vi*) refrigerator, (*vii*) furnace, (*viii*) bedroom apps, (*ix*) light plugs and (*x*) kitchen apps. The first four appliances

were classified as shiftable loads and the remaining ones considered as fixed loads. Neither User-1 nor User-2 had all ten appliances. More specifically, User-1 did not have bedroom apps or furnace and User-2 did not own an electric car.

The hourly temperature data for all sample days in $\mathcal{N}$ was also obtained from [93]. The hourly time of use rates were obtained from Austin Energy website [96]. The website provides electricity rates in the Austin area for a time of use pilot project. The user data used in this study were part of DER that received pricing information from the utility and were expected to defer their loads during peak or high energy price hours.

5.5.2 Load Disaggregation

The load disaggregation approach described in Section 5.4 was implemented separately on the total usage data for User-1 and User-2. The plots in Figures 5.3, 5.4 and 5.5 show the load disaggregation results obtained for User-1. All three figures contain the real loads (top, red) as well as the corresponding disaggregated quantities (bottom, blue) that were obtained by the proposed NMF-GMM hybrid algorithm. In Figure 5.3 are shown the shiftable loads, $\mathbf{X}^s$, and $\widetilde{\mathbf{X}}^s$ of User-1. The remarkable similarity between the two quantities is evident.

Figure 5.4 shows the corresponding fixed loads $\mathbf{X}^f$, and $\widetilde{\mathbf{X}}^f$. The aggregate loads, $\mathbf{X} = \mathbf{X}^f + \mathbf{X}^s$ and $\widetilde{\mathbf{X}} = \widetilde{\mathbf{X}}^f + \widetilde{\mathbf{X}}^s$ are provided in Figure 5.5. Among the most conspicuous discrepancies are at $t \approx 250, t \approx 660$, and $t \approx 1080$. At $t \approx 250{,}660$, a small fraction of the fixed load $\mathbf{X}^f$ is erroneously incorporated by the hybrid algorithm into the shiftable load $\widetilde{\mathbf{X}}^s$, whereas at $t \approx 1080$ some of $\mathbf{X}^s$ is transferred to $\widetilde{\mathbf{X}}^f$. However, these discrepancies are very small. The reverse inconsistencies are seen in Figure 5.5. However, they are exaggerated due to the smaller range of the fixed loads.

Figures 5.6, 5.7, and 5.8 are the corresponding loads associated with User-2. In this case, the most obvious difference occurs at $t \approx 470$, due to the appearance of a spike in the disaggregated shiftable load indicating that some fixed load was mislabeled as shiftable by the proposed algorithm. As before, this deviation is minor.

Table 5.1: Scaler utility parameter

	actual	computed
b	23.32	22.43

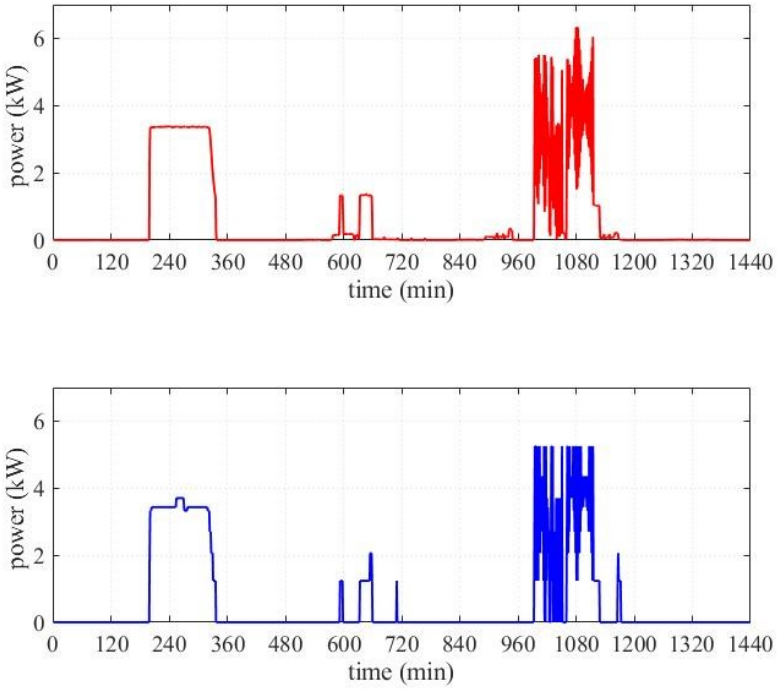

Figure 5.4: Shiftable load actual (top) and computed (bottom) for User-1

The disaggregated fixed and shiftable load results from Section 5.5.2 were used to compute the utility parameters using the parameter estimation approach proposed in Section 5.4.3. To validate the accuracy of the parameter estimation approach, utility parameters for actual fixed and shiftable loads are computed. The range and median values of utility parameters, a, d, x_0 for shiftable loads are presented in Table 5.2. They were computed for actual load and computed load data. Scaler parameter b is provided in Table 5.1. These parameter are also presented in Figures 5.10 – 5.12 for comparison. Simulation results validates the accuracy of the proposed model.

The plots in Figure 5.13 shows utility and cost curves for User-1 for one period.

Table 5.2: Load model parameters

	Range (actual)	Median(actual)	Range (computed)	Median (computed)
a	0 - 6.10	2.41	0 - 5.33	2.37
d	0 - 137.87	0	0 - 142.67	0
x_0	0 - 61.74	13.02	0 - 54.68	13.30

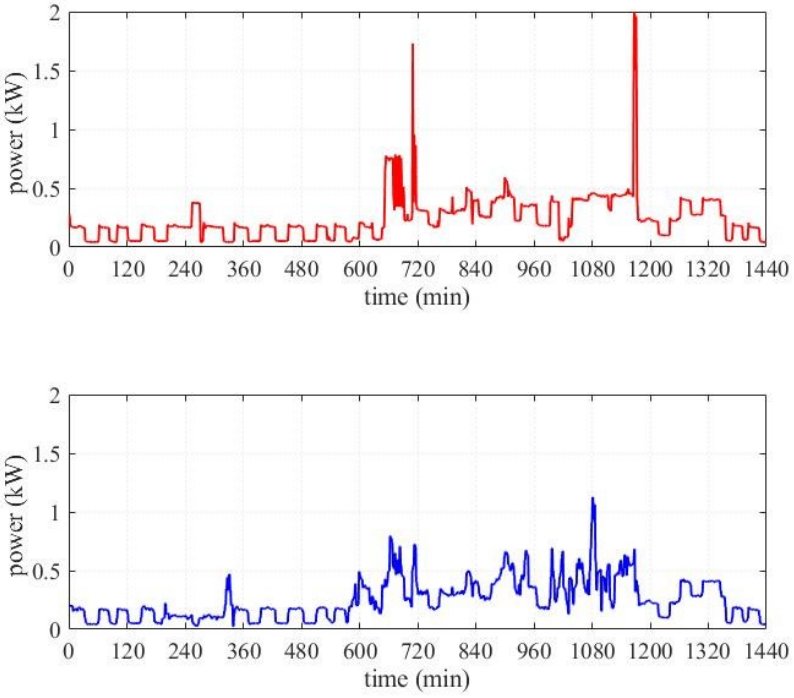

Figure 5.5: Fixed load actual (top) and computed (bottom) for User-1.

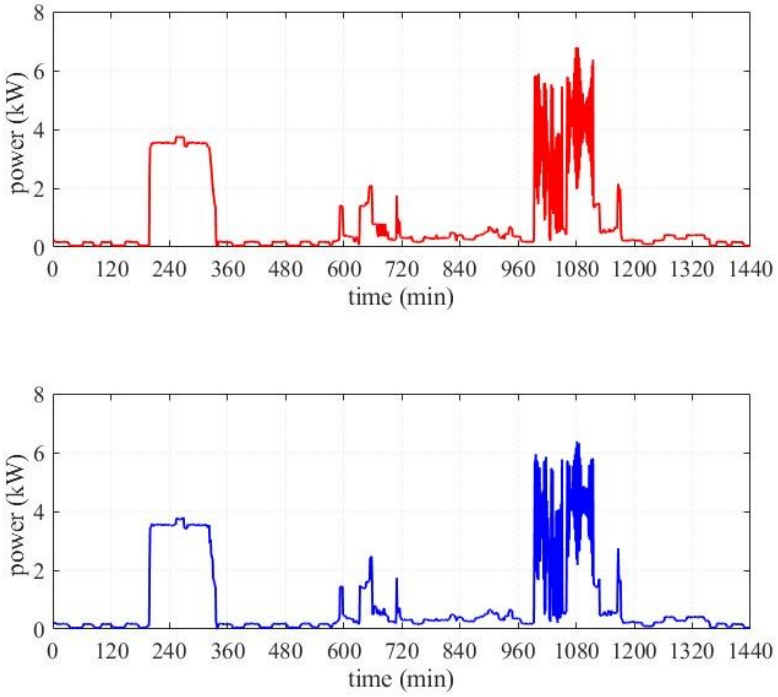

Figure 5.6: Aggregate load actual (top) and computed (bottom) for User-1.

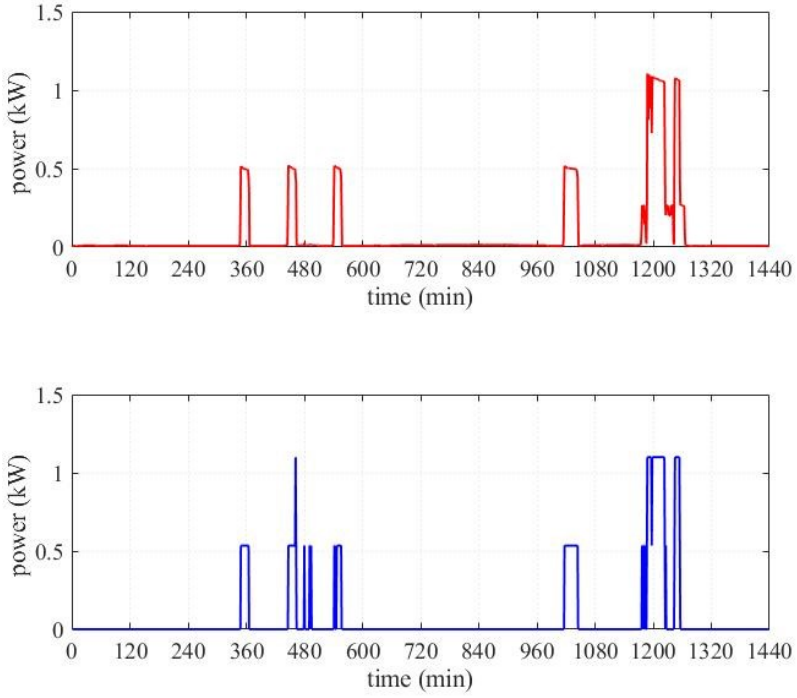

Figure 5.7: Shiftable load actual (top) and computed (bottom) for User-2.

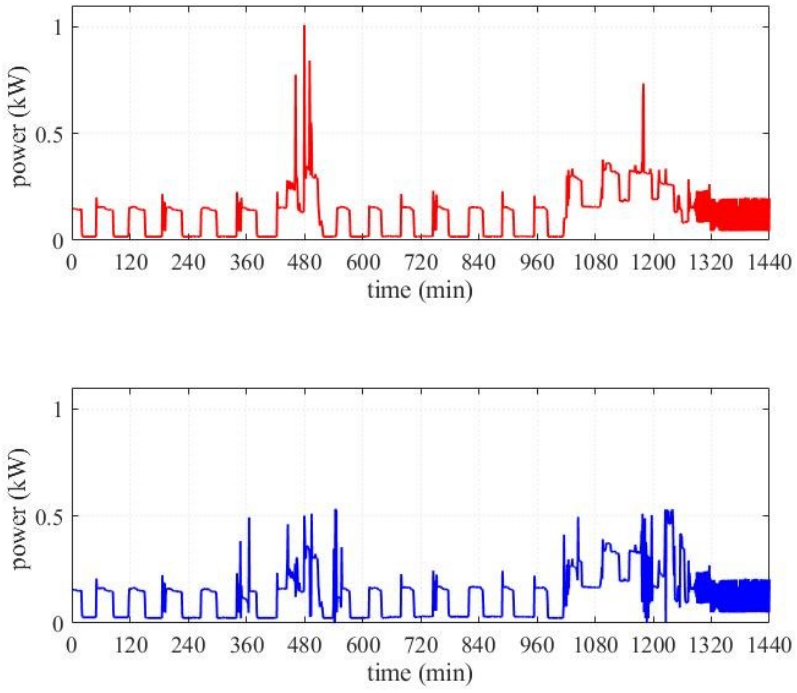

Figure 5.8: Fixed load actual (top) and computed (bottom) for User-2.

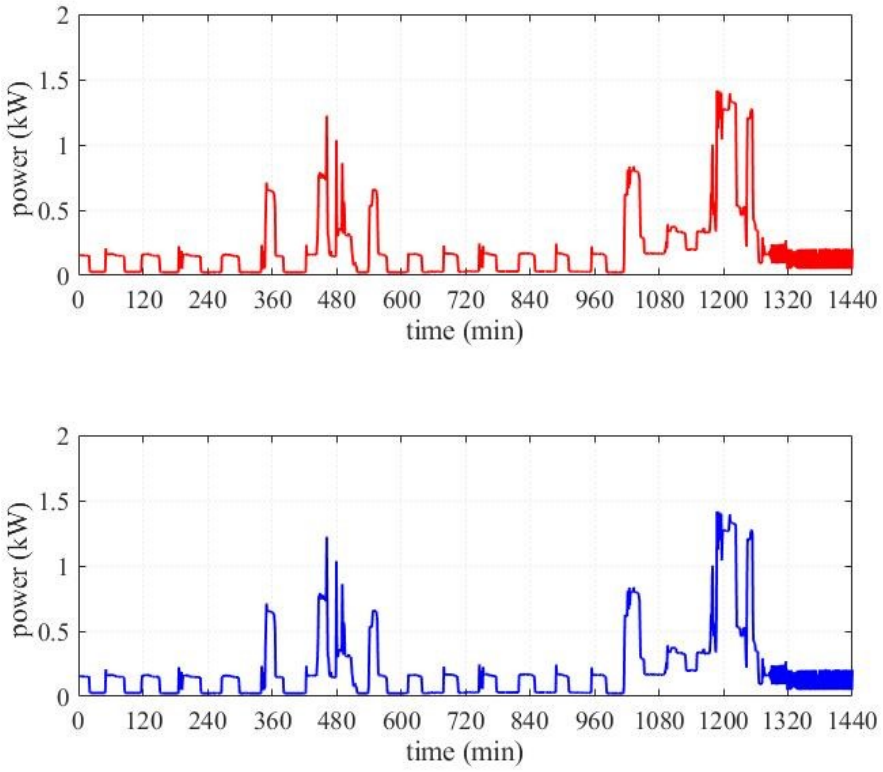

Figure 5.9: Aggregate load actual (top) and computed (bottom) for User-2.

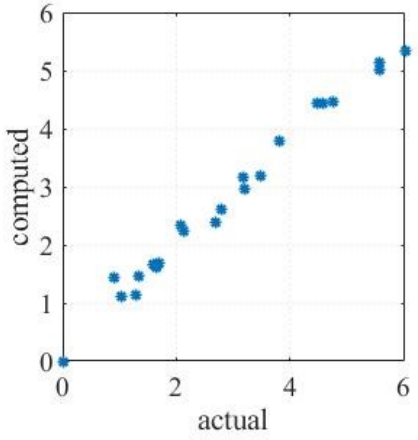

Figure 5.10: Actual vs. computed values of parameter a.

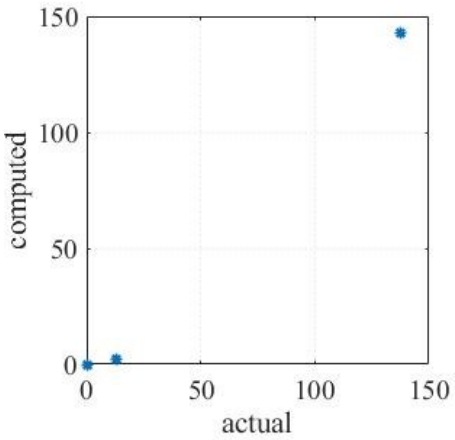

Figure 5.11: Actual vs. computed values of parameter d.

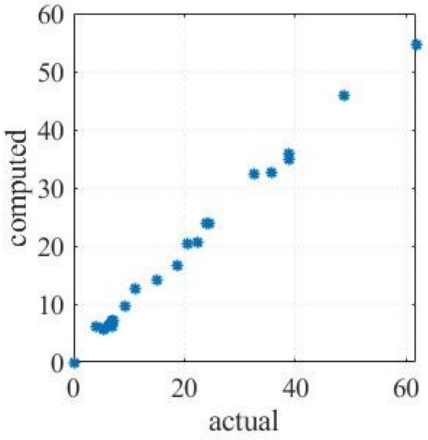

Figure 5.12: Actual vs. computed values of parameter x_0.

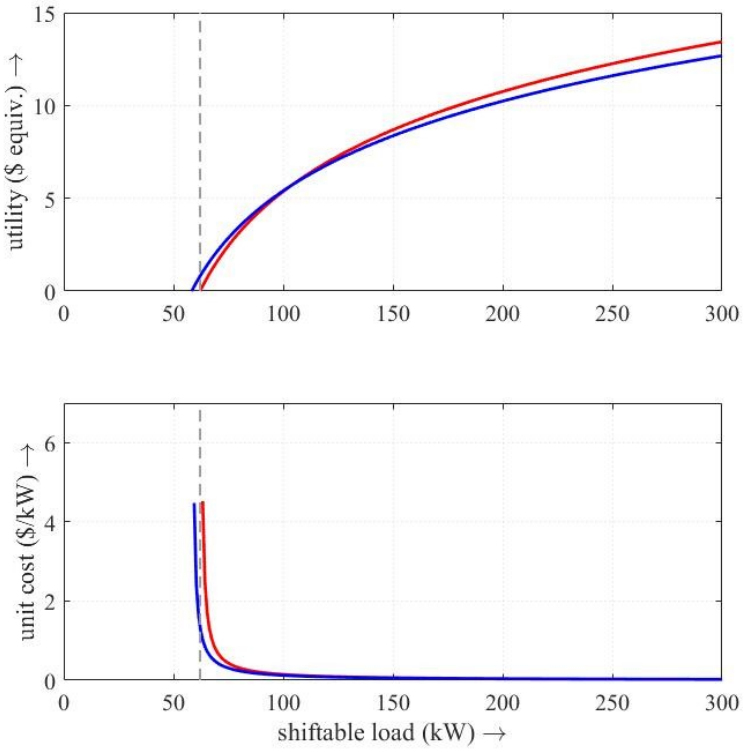

Figure 5.13: Utility curves(top), cost curves (bottom)

5.6 Conclusion

This research introduces a semi-parametric utility model that can be learned from smart-metering in an entirely non-intrusive manner. Although not discussed, parameter estimation can be accomplished using gradient descent or any other iterative algorithm, rendering the entire approach adaptive. There are many uses of such an adaptive utility model. It would allow demand response programs, energy trading, day-ahead planning, generator scheduling to proceed in an entirely automated manner without the need for human intervention.

In theory, it is possible to obtain an extensive utility model that is completely parametrized in terms of individual appliances. Each appliance can be represented with its own linear/nonlinear utility function; other factors besides temperature can be incorporated into it. This approach would require load disaggregation of a finer granularity. The GMM-NMF algorithm can be applied for this task, as it was observed by the authors that the loads consumed by individual appliances could be identified by visual inspection of the disaggregated loads. It remains to be seen whether the

larger errors that fine-grained disaggregation would be outweighted by the further flexibility it would provide in any application.

The proposed utility model is memoryless as the immediate past load history is disregarded. In other words, during each period *k*, the model does not take into account the actual load used earlier (x_{k-1}, x_{k-2}, etc.). Extending the utility model to consider prior usage, using ARMA, GARCH, Elman neural networks, or other forecasting tools would allow it to be used in load forecasting.

Chapter 6 - Conclusion

The research in this dissertation distinguishes between cost setting and power setting modes of DSO operation, introducing energy exchange mechanisms for each. These approaches are not only efficient and privacy-preserving, but also incorporate a fairness component in the form of the Jain's index of fairness. Chapters 2 and 3 establish theoretical results to show that the index is well suited for use in these mechanisms. Simulation results in both chapters, which were designed to supplement the theoretical analysis, corroborate the theoretical claims, thereby indicating that the Jain's index can effectively alleviate the effects of underlying locational disadvantages of DMLP in these auctions.

The effective use of primal and dual decompositions in the optimization algorithms allow the bilevel auctions to proceed seamlessly with the DSO operating under the power setting as well as the newer cost setting modes. The most significant difference between the studies in Chapters 3 and 4 are in their optimization framework. Whereas the optimization algorithm in Chapter 3 is based on classical convex optimization theory, Pareto-optimality theory serves as the backdrop for that in Chapter 4. It extends the a priori assumption of convexity that is typically assumed in the existing research literature on transactive energy trade, by showing that these mechanisms can operate with quasi-concave objectives.

Chapters 4 and 5 focus on a different, yet closely related aspect of energy auctions. Automation being the key enabling technology for such mechanisms, it is important for the service provider to glean as much information as possible through smart monitoring without undue intrusions into the privacies of individual residences. In this context, these chapters consider unsupervised learning approaches for energy disaggregation. Chapter 3 focuses on decomposing aggregate loads into individual appliances' energy consumption. On the other hand, Chapter 4 categorizes the load into two classes of appliances, using the disaggregated loads therein to estimate the parameters of consumer utility models. The remarkable similarities between the disaggregated loads with their real counterparts establish the effectiveness of the proposed machine learning algorithms.

There are three important directions along which future research can proceed.

- Pareto-optimality framework: It should be noted that although Chapter 4 uses vector optimization to achieve Pareto optimality, this was applied to a cost-setting mechanism. This

approach can be generalized for use in power-setting DSO mechanisms. As the latter uses dual decomposition, such investigations must rely on vector duality theory.

- Multiperiod auctions: The trading algorithms suggested in this research deals with single period auctions. The semi-parametric consumer modeling introduced in Chapters 4 and 5 can be leveraged to extend the auction mechanisms for multi-period use. The ability of the proposed disaggregation algorithms should be assessed and if necessary, extended further for online adaptation.
- Role of aggregators: The aggregators in the auction mechanisms were associated with intermediate nodes of the underlying grid so that its physical constraints could directly addressed at the DSO level. The proposed bilevel auctions can be extended for use under a more generalized setting where aggregators are detached from physical nodes, thereby offering them the flexibility of adopting a wide variety of pricing policies that satisfy the needs of their own sets of costumers.

References

[1] M. Nazif Faqiry and S. Das, "Distributed bilevel energy allocation mechanism with grid constraints and hidden user information," *IEEE Transactions on Smart Grid*, vol. 10, no. 2, pp. 1869–1879, 2019.

[2] S. Huang, Q. Wu, S. S. Oren, R. Li, and Z. Liu, "Distribution Locational Marginal Pricing Through Quadratic Programming for Congestion Management in Distribution Networks," *IEEE Transactions on Power Systems*, vol. 30, no. 4, pp. 2170–2178, 2015.

[3] H. Yuan, F. Li, Y. Wei, and J. Zhu, "Novel linearized power flow and linearized OPF models for active distribution networks with application in distribution LMP," *IEEE Transactions on Smart Grid*, vol. 9, no. 1, pp. 438–448, 2018.

[4] L. Bai, J. Wang, C. Wang, C. Chen, and F. Li, "Distribution Locational Marginal Pricing (DLMP) for Congestion Management and Voltage Support," *IEEE Transactions on Power Systems*, vol. 33, no. 4, pp. 4061–4073, 2018.

[5] S. Hanif, K. Zhang, C. Hackl, M. Barati, H. B. Gooi, and T. Hamacher, "Decomposition and Equilibrium Achieving Distribution Locational Marginal Prices using Trust-Region Method," *IEEE Transactions on Smart Grid*, vol. 3053, no. c, pp. 1–1, 2018.

[6] M. Caramanis, E. Ntakou, W. W. Hogan, A. Chakrabortty, and J. Schoene, "Co-optimization of power and reserves in dynamic T&D power markets with nondispatchable renewable generation and distributed energy resources," *Proceedings of the IEEE*, vol. 104, no. 4, pp. 807–836, 2016.

[7] Z. Liu, Q. Wu, S. Oren, S. Huang, R. Li, and L. Cheng, "Distribution Locational Marginal Pricing for Optimal Electric Vehicle Charging through Chance Constrained Mixed-Integer Programming," *IEEE Transactions on Smart Grid*, vol. 3053, no. c, pp. 1–1, 2016.

[8] S. Hanif, T. Massier, H. B. Gooi, T. Hamacher, and T. Reindl, "Cost Optimal Integration of Flexible Buildings in Congested Distribution Grids," *IEEE Transactions on Power Systems*, vol. PP, no. 99, pp. 1–14, 2016.

[9] S. Hanif, P. Creutzburg, H. B. Gooi, and T. Hamacher, "Pricing Mechanism for Flexible Loads using Distribution Grid Hedging Rights," *IEEE Transactions on Power Systems*, vol. 8950, no. c, pp. 1–12, 2018.

[10] Z. Baharlouei, M. Hashemi, H. Narimani, and H. Mohsenian-Rad, "Achieving optimality and fairness in autonomous demand response: Benchmarks and billing mechanisms," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 968–975, 2013.

[11] Z. Baharlouei and M. Hashemi, "Efficiency-fairness trade-off in privacy-preserving autonomous demand side management," *IEEE Transactions on Smart Grid*, vol. 5, no. 2. pp. 799–808, 2014.

[12] Z. Li, L. Chen, and G. Nan, “Small-scale Renewable Energy Source Trading: A Contract Theory Approach,” *IEEE Transactions on Industrial Informatics*, vol. 14, no. 4, pp. 1–1, 2017.

[13] O. Sener and V. Koltun, “Multi-task Learning As Multi-objective Optimization,” in *Proceedings of the 32Nd International Conference on Neural Information Processing Systems*, 2018, pp. 525–536.

[14] P. Jacquot, P. Jacquot, O. Beaude, S. Gaubert, and N. Oudjane, “Analysis and Implementation of an Hourly Billing Mechanism for Demand Response Management,” *IEEE Transactions on Smart Grid*, vol. 3053, no. c, pp. 1–14, 2018.

[15] J. Ma, J. Deng, L. Song, and Z. Han, “Incentive Mechanism for Demand Side Management in Smart Grid Using Auction,” *IEEE Trans. Smart Grid*, vol. 5, no. 3, pp. 1379–1388, 2014.

[16] S. K. Vuppala, S. Member, K. Padmanabh, and S. K. Bose, “Incorporating Fairness within Demand Response Programs in Smart Grid,” *Innovative Smart Grid Technologies (ISGT), IEEE PES*, pp. 1–9, 2011.

[17] A. Chis and V. Koivunen, “Coalitional game based cost optimization of energy portfolio in smart grid communities,” *IEEE Transactions on Smart Grid*, vol. 3053, no. c, pp. 1–11, 2017.

[18] M. Ghorbanian, H. Narimani, and G. R. Yousefi, “Billing mechanism design in an autonomous demand side management in a smart distribution network,” *2017 25th Iranian Conference on Electrical Engineering, ICEE 2017*, no. lCEE20 17, pp. 1284–1290, 2017.

[19] S. Das and B. K. Panigrahi, “Multi-Objective Evolutionary Algorithms,” in *Encyclopedia of Artificial Intelligence*, 2009.

[20] G. Aghajani and N. Ghadimi, “Multi-objective energy management in a micro-grid,” *Energy Reports*, vol. 4, pp. 218–225, 2018.

[21] A. Belgana, B. P. Rimal, and M. Maier, “Open Energy Market Strategies in Microgrids: A Stackelberg Game Approach Based on a Hybrid Multiobjective Evolutionary Algorithm,” *IEEE Transactions on Smart Grid*, vol. 6, no. 3, pp. 1243–1252, 2015.

[22] B. Zhou, R. Yang, C. Li, Y. Cao, Q. Wang, and J. Liu, “Multiobjective model of time-of-use and stepwise power tariff for residential consumers in regulated power markets,” *IEEE Systems Journal*, vol. 12, no. 3, pp. 2676–2687, 2018.

[23] S. S. Reddy, P. R. Bijwe, and A. R. Abhyankar, “Joint energy and spinning reserve market clearing incorporating wind power and load forecast uncertainties,” *IEEE Systems Journal*, vol. 9, no. 1, pp. 152–164, 2015.

[24] A. Ameli, M.-R. Farrokhifard, E. Davari-nejad, H. Oraee, and M.-R. Haghifam, “Profit-Based DG Planning Considering Environmental and Operational Issues: A Multiobjective Approach,” *IEEE Systems Journal*, vol. 11, no. 4, pp. 1959–1970, 2015.

[25] J. Cao, Z. Bu, Y. Wang, H. Yang, J. Jiang, and H.-J. Li, "Detecting Prosumer-Community Groups in Smart Grids From the Multiagent Perspective," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. PP, pp. 1–13, 2019.

[26] B. R. Sathyanarayana and G. T. Heydt, "Sensitivity-based pricing and optimal storage utilization in distribution systems," *IEEE Transactions on Power Delivery*, vol. 28, no. 2, pp. 1073–1082, 2013.

[27] N. Rezaei, A. Ahmadi, A. Khazali, and J. Aghaei, "Multiobjective Risk-Constrained Optimal Bidding Strategy of Smart Microgrids: An IGDT-Based Normal Boundary Intersection Approach," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 3, pp. 1532–1543, 2019.

[28] P. K. Shukla and K. Deb, "On finding multiple Pareto-optimal solutions using classical and evolutionary generating methods," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1630–1652, 2007.

[29] L. M. Graña Drummond and B. F. Svaiter, "A steepest descent method for vector optimization," *Journal of Computational and Applied Mathematics*, vol. 175, no. 2, pp. 395–414, 2005.

[30] J. A. Désidéri, "Multiple-gradient descent algorithm (MGDA) for multiobjective optimization," *Comptes Rendus Mathematique*, vol. 350, no. 5–6, pp. 313–318, 2012.

[31] J.-A. Desideri and R. Duvigneau, "Parametric optimization of pulsating jets in unsteady flow by Multiple-Gradient Descent Algorithm (MGDA)," in *Numerical Methods for Differential Equations, Optimization, and Technological Problems, Modeling, Simulation and Optimization for Science and Technology*, J. Périaux, W. Fitzgibbon, B. Chetverushkin, and O. Pironneau, Eds. 2017.

[32] O. Sener and V. Koltun, "Multi-Task Learning as Multi-Objective Optimization," no. NeurIPS, 2018.

[33] M. El Moudden and A. El Ghali, "Multiple reduced gradient method for multiobjective optimization problems," *Numerical Algorithms*, vol. 79, no. 4, pp. 1257–1282, Dec. 2018.

[34] J. Fliege and A. Vaz, "A Method for Constrained Multiobjective Optimization Based on SQP Techniques," *SIAM Journal on Optimization*, vol. 26, no. 4, pp. 2091–2119, 2016.

[35] M. A. T. Ansary and G. Panda, "A sequential quadratically constrained quadratic programming technique for a multi-objective optimization problem," *Engineering Optimization*, vol. 51, no. 1, pp. 22–41, 2019.

[36] V. Morovati and L. Pourkarimi, "Extension of Zoutendijk method for solving constrained multiobjective optimization problems," *European Journal of Operational Research*, vol. 273, no. 1, pp. 44–57, 2019.

[37] T. Tanino, T. Tanaka, M. Inuiguchi, M. Ehrgott, and D. M. Ryan, "The Method of Elastic

Constraints for Multiobjective Combinatorial Optimization and its Application in Airline Crew Scheduling," *Multi-Objective Programming and Goal Programming*, no. 2, pp. 117–122, 2013.

[38] B. Humala, S. N. A. U. Nambi, and R. V. Prasad, "UniversalNILM: A Semi-supervised Energy Disaggregation Framework using General Appliance Models," in *e-Energy*, 2018.

[39] F. C. C. Garcia, C. M. C. Creayla, and E. Q. B. Macabebe, "Development of an Intelligent System for Smart Home Energy Disaggregation Using Stacked Denoising Autoencoders," *Procedia Computer Science*, vol. 105, no. December 2016, pp. 248–255, 2017.

[40] I. H. Çavdar and V. Faryad, "New Design of a Supervised Energy Disaggregation Model Based on the Deep Neural Network for a Smart Grid," *Energies*, vol. 12, no. 7, p. 1217, 2019.

[41] S. Singh and A. Majumdar, "Deep sparse coding for non-intrusive load monitoring," *IEEE Transactions on Smart Grid*, vol. 9, no. 5, pp. 4669–4678, 2018.

[42] M. DrIncecco, S. Squartini, and M. Zhong, "Transfer Learning for Non-Intrusive Load Monitoring," *IEEE Transactions on Smart Grid*, vol. 3053, no. c, pp. 1–1, 2019.

[43] W. Kong, Z. Y. Dong, J. Ma, D. J. Hill, J. Zhao, and F. Luo, "An Extensible Approach for Non-Intrusive Load Disaggregation with Smart Meter Data," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 3362–3372, 2018.

[44] M. A. Mengistu, A. A. Girmay, C. Camarda, A. Acquaviva, and E. Patti, "A Cloud-Based On-Line Disaggregation Algorithm for Home Appliance Loads," *IEEE Transactions on Smart Grid*, vol. 10, no. 3, pp. 3430–3439, 2019.

[45] Z. Guo, Z. J. Wang, and A. Kashani, "Home appliance load modeling from aggregated smart meter data," *IEEE Transactions on Power Systems*, vol. 30, no. 1, pp. 254–262, 2015.

[46] W. Kong, Z. Y. Dong, D. J. Hill, J. Ma, J. H. Zhao, and F. J. Luo, "A Hierarchical Hidden Markov Model Framework for Home Appliance Modeling," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 3079–3090, 2018.

[47] P. A. Chou and R. I. Chang, "Unsupervised adaptive non-intrusive load monitoring system," *Proceedings - 2013 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2013*, pp. 3180–3185, 2013.

[48] A. Rahimpour, H. Qi, D. Fugate, and T. Kuruganti, "Non-Intrusive Energy Disaggregation Using Non-Negative Matrix Factorization with Sum-to-k Constraint," *IEEE Transactions on Power Systems*, vol. 32, no. 6, pp. 4430–4441, 2017.

[49] A. Miyasawa, Y. Fujimoto, and Y. Hayashi, "Energy disaggregation based on smart metering data via semi-binary nonnegative matrix factorization," *Energy and Buildings*, vol. 183, pp. 547–558, 2019.

[50] A. Miyasawa, M. Matsumoto, Y. Fujimoto, and Y. Hayashi, "Energy disaggregation based on semi-supervised matrix factorization using feedback information from consumers," *2017 IEEE PES Innovative Smart Grid Technologies Conference Europe, ISGT-Europe 2017 - Proceedings*, vol. 2018-Janua, pp. 1–6, 2018.

[51] M. Figueiredo, B. Ribeiro, and A. de Almeida, "Analysis of trends in seasonal electrical energy consumption via non-negative tensor factorization," *Neurocomputing*, vol. 170, pp. 318–327, 2015.

[52] T. Y. Ji *et al.*, "Non-Intrusive Load Monitoring Using Additive Factorial Approximate Maximum a Posteriori Based on Iterative Fuzzy C-Means," vol. 3053, no. 2018, pp. 1–11, 2019.

[53] K. He, D. Jakovetic, B. Zhao, V. Stankovic, L. Stankovic, and S. Cheng, "A Generic Optimisation-based Approach for Improving Non-intrusive Load Monitoring," *IEEE Transactions on Smart Grid*, vol. 10, no. 6, pp. 1–1, 2019.

[54] S. W. Park, L. B. Baker, and P. D. Franzon, "Appliance Identification Algorithm for a Non-Intrusive Home Energy Monitor Using Cogent Confabulation," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 714–721, 2019.

[55] K. He, L. Stankovic, J. Liao, and V. Stankovic, "Non-Intrusive Load Disaggregation Using Graph Signal Processing," *IEEE Transactions on Smart Grid*, vol. 9, no. 3, pp. 1739–1747, 2018.

[56] G. C. Koutitas and L. Tassiulas, "Low Cost Disaggregation of Smart Meter Sensor Data," *IEEE Sensors Journal*, vol. 16, no. 6, pp. 1665–1673, 2016.

[57] L. A. Barroso, A. Street, S. Granville, and M. V. Pereira, "Offering strategies and simulation of multi-item iterative auctions of energy contracts," *IEEE Transactions on Power Systems*, vol. 26, no. 4, pp. 1917–1928, 2011.

[58] N. Li, L. Chen, and S. H. Low, "Optimal demand response based on utility maximization in power networks," *IEEE Power and Energy Society General Meeting*, 2011.

[59] M. Li, H. Feng, F. Chen, and J. Kou, "Optimal versioning strategy for information products with behavior-based utility function of heterogeneous customers," *Computers & Operations Research*, vol. 40, no. 10, pp. 2374–2386, 2013.

[60] J. H. Choi, H. Ahn, and I. Han, "Utility-based double auction mechanism using genetic algorithms," *Expert Systems with Applications*, vol. 34, no. 1, pp. 150–158, 2008.

[61] G. Zachariadis and J. A. Barria, "Dynamic pricing and resource allocation using revenue management for multiservice networks," *IEEE Transactions on Network and Service Management*, vol. 5, no. 4, pp. 215–226, 2008.

[62] J. K. Kale, "Growth maximization and downside protection using power-log utility functions for optimizing portfolios with derivatives," *5th International Conference on*

Information Technology and Applications, ICITA 2008, no. Icita, pp. 282–286, 2008.

[63] J. Cox and R. Oaxaca, "Is bidding behavior consistent with bidding theory for private value auctions?," *Research in experimental economics*, vol. 6, pp. 131–148, 1996.

[64] A. Abdel-Hadi and C. Clancy, "A utility proportional fairness approach for resource allocation in 4G-LTE," *2014 International Conference on Computing, Networking and Communications, ICNC 2014*, pp. 1034–1040, 2014.

[65] T. Lan, D. Kao, M. Chiang, and A. Sabharwal, "An Axiomatic Theory of Fairness in Resource Allocation," *INFOCOM (Extended)*, pp. 1–9, 2010.

[66] T. Hobfeld, L. Skorin-Kapov, P. E. Heegaard, and M. Varela, "Definition of QoE Fairness in Shared Systems," *IEEE Communications Letters*, vol. 21, no. 1, pp. 184–187, 2017.

[67] R. V. Prasad, E. Onur, and I. G. M. M. Niemegeers, "Fairness in Wireless Networks:Issues, Measures and Challenges," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 5–24, 2014.

[68] F. Chiti, R. Fantacci, and B. Picano, "A Matching Theory Framework for Tasks Offloading in Fog Computing for IoT Systems," *IEEE Internet of Things Journal*, vol. PP, no. c, pp. 1–1, 2018.

[69] C. Guo, S. Member, M. Sheng, X. Wang, and Y. Zhang, "Throughput Maximization with Short-Term and Long-Term Jain ' s Index Constraints in Downlink OFDMA Systems," *IEEE Transactions on Communication*, vol. 62, no. 5, pp. 1503–1517, 2014.

[70] C. L. Chang and J. C. H. Peng, "A Decision-Making Auction Algorithm for Demand Response in Microgrids," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 3553–3562, 2018.

[71] M. N. Faqiry and S. Das, "Double Auction With Hidden User Information: Application to Energy Transaction in Microgrid," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–14, 2018.

[72] A. K. Zarabie and S. Das, "Efficient Distributed DSO Auction with Linearized Grid Constraints," in *2019 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, 2019, pp. 1–5.

[73] M. N. Faqiry and S. Das, "Double-Sided Energy Auction in Microgrid: Equilibrium under Price Anticipation," *IEEE Access*, vol. 4, pp. 3794–3805, 2016.

[74] M. N. Faqiry and S. Das, "Distributed bilevel energy allocation mechanism with grid constraints and hidden user information," *IEEE Transactions on Smart Grid*, vol. 10, no. 1869–1879, 2019.

[75] C. Joe-Wong, S. Sen, T. Lan, and M. Chiang, "Multi-resource allocation: Fairness-efficiency tradeoffs in a unifying framework," in *2012 Proceedings IEEE INFOCOM*, 2012,

pp. 1206–1214.

[76] G. Fortino *et al.*, "Optimal Dynamic Pricing for Trading-Off User Utility and Operator Profit in Smart Grid," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–13, 2017.

[77] K. Miettinen, *Nonlinear multiobjective optimization*, vol. 12. Springer Science & Business Media, 2012.

[78] S. F. Richard and W. R. Zame, "Proper preferences and quasi-concave utility functions," *Journal of Mathematical Economics*, vol. 15, no. 3, pp. 231–247, Jan. 1986.

[79] A. K. Zarabie, S. Das, and M. N. Faqiry, "Fairness-Regularized DLMP-Based Bilevel Transactive Energy Mechanism in Distribution Systems," *IEEE Transactions on Smart Grid*, vol. 10, no. 6, pp. 6029–6040, 2019.

[80] N. O. Da Cunha and E. Polak, "Constrained minimization under vector-valued criteria in finite dimensional spaces," *Journal of Mathematical Analysis and Applications*, vol. 19, no. 1, pp. 103–124, 1967.

[81] A. Bin Sediq, R. H. Gohary, R. Schoenen, and H. Yanikomeroglu, "Optimal tradeoff between sum-rate efficiency and jain's fairness index in resource allocation," *IEEE Transactions on Wireless Communications*, vol. 12, no. 7, pp. 3496–3509, 2013.

[82] K. D. Bae and D. S. Kim, "Optimality and Duality Theorems in Nonsmooth Multiobjective Optimization," *Fixed Point Theory and Applications*, vol. 2011, no. 1, p. 42, Aug. 2011.

[83] A. K. Zarabie and S. Das, "Efficient Distributed DSO Auction with Linearized Grid Constraints," in *10th Conference on Innovative Smart Grid Technologies(ISGT)*, 2019, pp. 1–5.

[84] K. Jain, V. V Vazirani, and Y. Ye, "Market Equilibria for Homothetic, Quasi-concave Utilities and Economies of Scale in Production," in *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2005, pp. 63–71.

[85] K. J. Arrow and A. C. Enthoven, "Qusi-Concave Programming," *Econometrica*, vol. 29, no. 4, pp. 779–800, 1961.

[86] D. Bertsimas, V. F. Farias, and N. Trichakis, "On the Efficiency-Fairness Trade-off," *Management Science*, vol. 58, no. 12, pp. 2234–2250, 2012.

[87] M. Matsumoto, Y. Fujimoto, and Y. Hayashi, "Energy Disaggregation Based on Semi-Binary NMF," in *MLDM*, 2016.

[88] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[89] D. D. Lee, M. Hill, and H. S. Seung, "Algorithms for Non-Negative Matrix Factorization,"

in *Proceeding NIPS'00 Proceedings of the 13th International Conference on Neural Information Processing Systems*, 2000, no. 1, pp. 535–541.

[90] N. Gillis, "The Why and How of Nonnegative Matrix Factorization," *ArXiv*, pp. 1–25, 2014.

[91] J. Baglama, L. Reichel, and D. Richmond, "An Augmented LSQR Method," *Numer. Algorithms*, vol. 64, no. 2, pp. 263–293, Oct. 2013.

[92] R. Peharz and F. Pernkopf, "Sparse nonnegative matrix factorization with ℓ 0-constraints," *Neurocomputing*, vol. 80. pp. 38–46, 2012.

[93] "Pecan Street Inc. Dataport." [Online]. Available: https://www.pecanstreet.org/.

[94] A. S. W. A. Barnett, "Quantifying Consumer Preferences Chapter 1: Measuring Consumer Preferences and Estimating Demand Systems," in *Contributions to Economic Analysis*, 2009, pp. 1–35.

[95] J.-H. Yoo and S.-J. Choi, "Nonnegative Matrix Factorization with Orthogonality Constraints," *Journal of Computing Science and Engineering*, vol. 4, no. 2, pp. 97–109, 2010.

[96] "Asuting Energy." [Online]. Available: https://austineenergy.com/ae/rates/. [Accessed: 09-Mar-2019].

Appendix A - Nomenclature

Nomenclature for Chapter 2

$\mathcal{N}$	Set of nodes, lines $\lvert\mathcal{N}\rvert = N$
$\mathcal{A}$	Set of nodes with aggregators, $\lvert\mathcal{A}\rvert = A$
k, l	Node indices, $k, l \in \mathcal{N}$
$u(k)$	Index of k's immediate upstream (parent) node
$d(k)$	Set of k's immediate downstream nodes
r_k, x_k	Resistance, reactance of line $(u(k), k)$, $k \in \mathcal{N}$
p_k, q_k	Real, reactive power injection into node, $k \in \mathcal{A}$
c_k	Per unit cost of node $k \in \mathcal{A}$
$P_k, Q_k,$	Real, reactive line power flows $k \in \mathcal{N}$
L_k^P, L_k^Q	Real, reactive losses $k \in \mathcal{N}$
V_k, δ_k	Voltage, angle of node $k \in \mathcal{A}$
ϵ	Maximum allowable pu voltage deviation
$u(\cdot)$	Immediate upstream node
$d(\cdot),$	Set of immediate downstream nodes
$\mathcal{U}(\cdot),$	Set of all upstream nodes of given node
$\mathbf{D}, \mathbf{T}$	Downstream and tree matrices
$\mathcal{G}_k$	Set of agents at aggregator$k$, $k \in \mathcal{A}$, $G_k = \lvert\mathcal{G}_k\rvert$
i	Index of agent $i \in \mathcal{G}_k$
p_k^i, q_k^i	Real and reactive demands of $i \in \mathcal{G}_k$
g_k^i, a_k^i, b_k^i	Generation and utility parameters of $i \in \mathcal{G}_k$
$\underline{\boldsymbol{\alpha}}, \overline{\boldsymbol{\alpha}}, \boldsymbol{\beta}, \lambda, \gamma$	Dual variables
$\mathbf{c}_C, \mathbf{c}_V$	DLMP components for congestion, voltage
$\mathbf{c}_{E+L}, \mathbf{c}_F$	DLMP components for energy and loss, fairness
$\Omega(\cdot)$	System level objective function
$\mathcal{W}_k(\cdot)$	Social welfare of aggregator $k \in \mathcal{A}$
$u_k^i(\cdot)$	Utility of $i \in \mathcal{G}_k$
$\pi_k^i(\cdot)$	Payoff of $i \in \mathcal{G}_k$

$\mathfrak{L}(\cdot)$	Lagrange function
$J(\cdot)$	Jain's index of fairness
η	Increment factor per iteration
C	Regularization weight

Nomenclature for Chapter 3

$\mathcal{N}$	Set of nodes in grid, $\lvert\mathcal{N}\rvert = N$
$\mathcal{A}$	Subset of nodes with aggregators, $\lvert\mathcal{A}\rvert = A$
k	Node index, $k \in \mathcal{N}$
p_k	Power injection into node, $k \in \mathcal{A}$
c_k	Per unit cost of node $k \in \mathcal{A}$
$\mathcal{G}_k$	Set of agents at aggregator$k$, $k \in \mathcal{A}$, $G_k = \lvert\mathcal{G}_k\rvert$
$\mathcal{G}_k^{\mathrm{p}}, \mathcal{G}_k^{\mathrm{C}}$	Producer and consumer subsets of $\mathcal{G}_k$
i	Index of agent $i \in \mathcal{G}_k$
p_k^i	Energy demand of agent $i \in \mathcal{G}_k$
g_k^i, a_k^i, b_k^i	Generation and utility parameters of $i \in \mathcal{G}_k$
P_0	Total energy at DSO
c_0	Unit cost at DSO
$\mathbf{p}_k$	$G_k \times 1$ energy vector at aggregator $k \in \mathcal{A}$
$\mathbf{c}$	$A \times 1$ unit cost vector
$\mathbf{p}$	$A \times 1$ energy allocation vector
$\mathbf{C}^V, \mathbf{c}_l^V, \mathbf{c}_u^V$	Voltage constraint related constants
$\mathbf{C}^S, \mathbf{c}_0^S$	Line capacity related constraint constants
$\mathbf{c}^{P_0}, c_0^{P_0}, \mathbf{c}_0^S$	Energy balance condition related constants
$\boldsymbol{\Omega}(\cdot)$	Vector objective function
$\mathcal{W}(\cdot)$	Global welfare function
$\mathcal{R}(\cdot)$	Global fairness function
$J(\cdot)$	Jain's index of fairness
$\mathcal{U}_k(\cdot)$	Aggregated utility of aggregator $k \in \mathcal{A}$

Appendix B - Linearized Constraints

With the exception of the budget balance constraint Eqn. (3.23*f*) in Eqn. (3.23), which is straightforward to obtain, the expressions associated with all other constants involved in the remaining constraints in Eqn. (3.23*b*) – (23*e*), are provided below.

(*i*) Voltage constraint

$$\mathbf{C}^V = \mathbf{C}_p^V + \mathbf{C}_q^V \boldsymbol{\Theta},$$
$$\mathbf{c}_l^V = -\mathbf{c}_0^V + \mathbf{1}_N V_0 - \boldsymbol{\epsilon},$$
$$\mathbf{c}_u^V = \mathbf{c}_0^V - \mathbf{1}_N V_0 - \boldsymbol{\epsilon}.$$

The vector $\mathbf{c}_0^V$ and the matrices $\mathbf{C}_p^V$ and $\mathbf{C}_q^V$ in the above expression are,

$$\mathbf{c}_0^V = \left[\mathbf{C}\begin{bmatrix}\mathbf{T}+\mathbf{I} & \mathbf{0}\\ \mathbf{0} & \mathbf{T}+\mathbf{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{L}_0^P - \mathbf{J}_P^{L\,\mathrm{T}}\mathbf{p}_0 - \mathbf{J}_P^{L\,\mathrm{T}}\mathrm{diag}(\boldsymbol{\theta})^{-1}\mathbf{q}_0\\ \boldsymbol{L}_0^Q - \mathbf{J}_Q^{L\,\mathrm{T}}\mathbf{p}_0 - \mathbf{J}_Q^{L\,\mathrm{T}}\mathrm{diag}(\boldsymbol{\theta})^{-1}\mathbf{q}_0\end{bmatrix} - \mathbf{C}\begin{bmatrix}\mathbf{B}^r V_0\mathbf{e} + \mathbf{B}^x\delta_0\mathbf{e}\\ \mathbf{B}^x V_0\mathbf{e} + \mathbf{B}^r\delta_0\mathbf{e}\end{bmatrix}\right]_{r=1:N},$$

$$\mathbf{C}_p^V = \left[\mathbf{C}\begin{bmatrix}(\mathbf{T}+\mathbf{I})\mathbf{A} & \mathbf{0}\\ \mathbf{0} & (\mathbf{T}+\mathbf{I})\mathbf{A}\end{bmatrix}\right]_{r=1:N,c=1:A},$$

$$\mathbf{C}_q^V = \left[\mathbf{C}\begin{bmatrix}(\mathbf{T}+\mathbf{I})\mathbf{A} & \mathbf{0}\\ \mathbf{0} & (\mathbf{T}+\mathbf{I})\mathbf{A}\end{bmatrix}\right]_{r=1:N,c=A+1:2A}.$$

(*ii*) Flow constraint

$$\mathbf{C}^S = \mathbf{C}_p^P + \mathbf{C}_p^Q + \left(\mathbf{C}_q^P + \mathbf{C}_q^Q\right)\boldsymbol{\Theta},$$
$$\mathbf{c}_0^S = \mathbf{c}_0^P + \mathbf{c}_0^Q - \sqrt{2}\boldsymbol{S}.$$

The vector $\mathbf{c}_0^P$ and the matrices $\mathbf{C}_p^P$ and $\mathbf{C}_q^P$ appearing in the above expressions are,

$$\mathbf{C}_p^P = \left(\mathbf{A} + \mathbf{T}\mathbf{A} + \mathbf{T}\mathbf{J}_P^{L\,\mathrm{T}}\right),$$
$$\mathbf{C}_q^P = \mathbf{T}\mathbf{J}_P^{L\,\mathrm{T}}\mathrm{diag}(\boldsymbol{\theta})^{-1},$$
$$\mathbf{c}_0^P = \mathbf{T}\left(\boldsymbol{L}_0^P - \mathbf{J}_P^{L\,\mathrm{T}}\mathbf{p}_0 - \mathbf{J}_P^{L\,\mathrm{T}}\mathrm{diag}(\boldsymbol{\theta})^{-1}\mathbf{q}_0\right).$$

(*iii*) Power balance

$$\mathbf{c}^{P_0} = \boldsymbol{C}_p^{P_0} + \boldsymbol{\Theta}\boldsymbol{C}_q^{P_0},$$

$$c_0^{P_0} = \mathbf{1}_N^{\mathrm{T}} \left(\boldsymbol{L}_0^P - \mathbf{J}_P^{L^{\mathrm{T}}} \mathbf{p}_0 - \mathbf{J}_P^{L^{\mathrm{T}}} \mathrm{diag}(\boldsymbol{\theta})^{-1} \mathbf{q}_0 \right).$$

In the above expression, the vectors $\boldsymbol{C}_p^{P_0}$ and $\boldsymbol{C}_{pq}^{P_0}$ are given by the following pair of equations,

$$\boldsymbol{C}_p^{P_0} = \mathbf{1}_A^{\mathrm{T}} + \mathbf{1}_N^{\mathrm{T}} \mathbf{J}_P^{L^{\mathrm{T}}},$$

$$\boldsymbol{C}_q^{P_0} = \mathbf{1}_N^{\mathrm{T}} \mathbf{J}_P^{L^{\mathrm{T}}} \mathrm{diag}(\boldsymbol{\theta})^{-1}.$$