UNIVERSITÀ DEGLI STUDI DI CAGLIARI

DIPARTIMENTO DI MATEMATICA E INFORMATICA
DOTTORATO DI RICERCA IN MATHEMATICS AND COMPUTER SCIENCE
CICLO XXXII

PH.D. THESIS

Mathematical Models for Improving Flexibility within the Smart Grid domain

S.S.D. INF/01

CANDIDATE
Fabio Lilliu

SUPERVISOR
Prof. Diego Reforgiato Recupero

PHD COORDINATOR
Prof. Michele Marchesi

Final examination academic year 2018/2019
July 7, 2020

Abstract

One of the most important topics of the last decades has been finding energy sources who can replace fossil fuels. Renewable energy is a good candidate, being virtually inexhaustible and more environment-friendly. In order to allow for this transition, electric energy grids have evolved and have become new objects, called *smart grids*. However, the complexity of this new type of grid brings new issues and challenges, which are currently object of study for many researchers.

The purpose of this thesis is to showcase some of these problems, and to build mathematical models and algorithms in order to solve them, by leveraging a new property relative to energy loads: flexibility. Since smart devices are becoming more common and they can be remotely controlled, manipulation of energy profiles is possible, and this is a powerful tool for the management of smart grids.

Going more into detail, a framework for managing demand response, peak shaving and energy trading has been designed by the means of a combinatorial approach, and it has been enhanced by exploitation of parallel computing. Moreover, an incentive mechanism for usage of renewable energy has been analyzed and improved, by changing some functions which define its behavior. This mechanism has also been examined from a game-theoretic point of view, and it has been further improved in order to always guarantee an agreement between users for flexibility usage. Finally, a decentralized, multi-agent system approach has been used to solve the problems of cost optimization and congestion management. Most of the content of this thesis derives from research works published in journals and conferences.

Acknowledgements

This thesis concludes a work of three years. It began when, after obtaining my M.Sc. in Mathematics, I changed completely field of study and started a Ph.D. in Computer Science focused on smart grids, a completely new topic for me. During this journey, I met many people who taught a lot to me, in several aspects: technical, research-related, and human. Although I still have a lot of learn, this experience allowed me to grow both professionally and as a person, and for this reason I wanted to express my gratitude to these people.

First of all, I would like to thank my supervisor, Prof. Diego Reforgiato Recupero, for too many reasons to be counted. He has guided me during these years, allowed me to exploit my strong points and helped me correcting my weaknesses, helped me becoming more independent in my research work but never left me alone at the same time. In particular, he taught me how to be a researcher: although I am still very inexperienced, everything I learned on this aspect, I owe it to him. I could not have asked for a better guidance during my Ph.D.

I would also like to express my gratitude for my two supervisors in my abroad period: Dr. Meritxell Vinyals and Dr. Roman Denysiuk. I really learned a lot from them, especially in the smart grid field, and I appreciated a lot the possibility to work with them: in our team everyone had different expertises, and I feel like this diversity has been our strength. I grew a lot thanks to them.

Furthermore, I want to thank my colleagues both from my university and from my abroad experience: for their kindness, for their company, for having made me feel at home when in a foreign country, for everything I learned from them. I have been happy to meet you all.

Outside of the work environment, I would like to thank my family. In particular I want to thank Marcello and Ada, my parents, and Luca, my brother, for their invaluable support and for being always there for me, even when we are far apart. Their presence has been very important for me during these years, and I owe them a lot.

Finally, I want to thank all my friends: the ones I know since a lifetime and the ones I met during this journey, the ones I've seen countless times and the ones I could only meet online. Their support and their presence have been really meaningful for me during these years.

Publications

The research reported in this thesis has contribuited to the following publications:

- [1] Lilliu F., Loi A., Reforgiato Recupero D., Sisinni M., & Vinyals M. (2019). An Uncertainty-Aware Optimization Approach For Flexible Loads Of Smart Grid Prosumers: a Use Case on the Cardiff Energy Grid. Sustainable Energy, Grids and Networks, 20, (art. 100272).

- [2] Lilliu F., Vinyals M., Denysiuk R., & Reforgiato Recupero D. (2019). A novel payment scheme for trading renewable energy in smart grid. In proceedings of ACM e-Energy 2019 (pp.111-115).

- [3] Denysiuk R., Lilliu F., Reforgiato Recupero D., & Vinyals M. (2020). Peer-to-peer Energy Trading for Smart Energy Communities. In proceedings of ICAART 2020 (pp.40-49).

- [4] Denysiuk R., Lilliu F., Reforgiato Recupero D., & Vinyals M. (2020). Multiagent System for Community Energy Management. In proceedings of ICAART 2020 (pp.28-39).

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Context and contributions

In the last decades, the topic of energy is becoming more and more important. There are serious problems related to fossil fuels, as they are destined to run out at some point in future, and they are related to environmental issues such as climate change. For this reason, more attention is being given to other energy sources, in particular renewable ones, which address effectively these matters [5]. One key aspect of renewable energy is the fact that its production does not need to be centralized anymore: medium private producers, even single households, can become energy producers. This brought a significant change in the organization of electric energy grids, and a new type of structure has been born as a consequence of this, which has been named **smart grid**.

A smart grid is an electric energy grid which makes use of communication technology for its functioning and management. This allows the grid to make use of smart devices, and to collect and manage important data on consumption of the grid users. The most remarkable feature of a smart grid is the fact that the energy does not need to flow one-way anymore, from the energy provider to the grid users: this means that grid users have the possibility to produce energy, for example with photovoltaic systems, and inject it into the grid. For this reason, smart grids are extremely important in the context of renewable energy: thanks to their characteristics, grid users have the possibility to produce renewable energy for their own sustaining, and also to be rewarded for that, as they are paid for the energy they inject into the grid [6, 7].

Being much more complex than ordinary energy grids, smart grids present new issues that have to be taken into account. The introduction of decentralized production of renewable energy is problematic for grid stability, as energy production is not controllable and variable, and predictive models always have some error margin [8]. Also, the fact that some types of energy (such as solar) can only be produced at certain times of the day means that production may be scarce at certain times, and excessive at others. In addition to that, producing too much energy at the same time risks to create problems to the grid: such an event is called a congestion. Furthermore, the fact that users can produce and sell energy can enable them to enter the local energy market [9]. Such a market needs to be given precise rules, in order to avoid undesired behaviors which can be beneficial to certain parties in economic terms, but may be harmful for other users or for the well-functioning of the grid itself.

The technology exploited in smart grids permits a much more detailed and effective management of the grid itself. In particular, smart meters and smart devices allow for i) monitoring precisely consumption and production of energy through time, at device level; and ii) allowing the energy profiles of these devices to be shifted, curtailed or changed in time, according to the grid actors' needs. This latter property for energy loads is called **flexibility** [10], and plays a crucial role in the solution of the aforementioned issues for smart grids. For this reason, an important part of research on smart grid is centered on this theme.

This thesis shows how flexibility can be used to tackle several well-known questions related to the smart grid environment. It shows how the properties of flexibility can be exploited by means of mathematical models to maximise the profit for the multiple users, the stability of the grid and/or the energy self-consumption. Among all the open issues existing for smart grids, these are the ones that have been addressed in this work:

1. Managing demand response, enabling users to participate to the energy market and to obtain profits by selling their flexibility to agents interested to buy it.

2. Identifying and addressing the potential limitations of a state-of-the-art renewable energy incentive mechanism, such as the promotion of energy production curtailment or the lack of self-consumption guarantee

3. Analyzing the behavior of the grid users in the context of a renewable energy incentive mechanism, and guaranteeing the existence of a scenario which is the most profitable for all of them.

4. Optimizing the costs for the users of a local energy community, and using flexibility to avoid congestions in the grid, both in the case of excessive production and in the case of excessive consumption.

In order to face these problems, different models, strategies and algorithms have been elaborated, and will be detailed in this thesis. To be more precise, this thesis makes the following contributions to the state of the art:

1. Elaboration of a combinatorial optimization framework for smart grids, realized with exploitation of CUDA architecture for parallel computing, with the purpose of reducing costs and discomfort for grid users, managing the internal energy market of the grid and processing requests for demand response.

2. Improvement of an incentive system for renewable energy based on a virtual currency, highlighting its weak points and designing for it new price functions in order to overcome these weaknesses.

3. Analysis of the aforementioned incentive system through a game theory perspective, with the purpose of analyzing the behavior of each user depending on the other users' actions, and designing new price functions which guarantee an agreement between the users.

4. Creation of a multiagent system framework for managing a smart grid, which has the purpose of optimizing costs for grid users and managing congestions at local grid level. We show how this approach scales very well with the size of the grid thanks to its decentralized architecture.

1.2 Contents of the thesis

The rest of the thesis is organized as follows.

- Chapter 2 contains preliminary concepts related to the smart grid context and the mathematical background used in the rest of this thesis.

- Chapter 3 presents the problems of demand response management, peak shaving, cost optimization and management of the energy market in a grid, and showcases a three-steps optimization algorithm which solves sequentially these problems. This work has been done in cooperation with Professor Diego Reforgiato (*University of Cagliari*), Professor Andrea Loi (*University of Cagliari*), Dr. Mario Sisinni (*R2M Solution s.r.l.*) and Dr. Meritxell Vinyals (*CEA - Commissariat à l'énergie atomique et aux énergies alternatives*). This work has been published in [1].

- Chapter 4 contains the description of an incentive system for renewable energy called NRG-X-Change, an analysis of its weaknesses and the proposal of two new pricing functions in order to make this system more robust. This work has been done in cooperation with Professor Diego Reforgiato (*University of Cagliari*), Dr. Roman Denysiuk (*CEA - Commissariat à l'énergie atomique et aux énergies alternatives*) and Dr. Meritxell Vinyals (*CEA - Commissariat à l'énergie atomique et aux énergies alternatives*). This work has been published in [2].

- Chapter 5 focuses on the same incentive mechanism of the previous chapter, but analyzes it from a game theory perspective. It shows some sufficient conditions for reaching a Nash equilibrium, and creates some new pricing functions which respect said conditions. This work has been done in cooperation with Dr. Roman Denysiuk (*CEA - Commissariat à l'énergie atomique et aux énergies alternatives*), Professor Diego Reforgiato (*University of Cagliari*) and Dr. Meritxell Vinyals (*CEA - Commissariat à l'énergie atomique et aux énergies alternatives*). This work has been published in [3].

- Chapter 6 proposes a multiagent system approach to manage a smart grid and solve the problems of cost optimization and congestion management; the system is decentralized and is optimized with an algorithm named ADMM. This work has been done in cooperation with Dr. Roman Denysiuk (*CEA - Commissariat à l'énergie atomique et aux énergies alternatives*), Professor Diego Reforgiato (*University of Cagliari*) and Dr. Meritxell Vinyals (*CEA - Commissariat à l'énergie atomique et aux énergies alternatives*). This work has been published in [4].

- Chapter 7 shows the conclusions reached for each of the analyzed problems, the results obtained and the possible development of research in these fields.

Chapter 2

Preliminary concepts

2.1 Smart grids

Smart grids are much more complex structures compared to the pre-existing energy grids; for this reason, new figures and concepts have been defined in this context. In this section, we will describe some of the key terms that will be used in this thesis.

2.1.1 Smart Grid concepts

In this subsection we describe some of the most important concepts that will be utilized in the following chapters. They refer to properties relative to actions that can be taken in a smart grid environment, and have to be taken carefully into account when working in this context.

- **Flexibility.** This term is the keystone concept of this thesis, and indicates the possibility of a grid user to alter their energy profile. Different devices may offer different types of flexibility: for example, an appliance such as a dishwasher or a washing machine has a defined energy consumption profile which can be shifted in time, while a heat pump may have its power output increased or decreased at certain times, depending on the needs of the user.

- **Congestion.** A congestion happens when the grid is overloaded with energy: this is a dangerous situation for the grid, and may lead to malfunctioning of the structure and to lasting damage on it. There are two principal causes for congestion: excess of consumption request from the users (**overconsumption**), and excess of energy production through the grid (**overproduction**).

- **Curtailment.** An energy load is said to be curtailed if its amount is significantly reduced on purpose. This term may refer to consumption as well as production. Consumption curtailment happens for example when the user decreases the power output of a flexible device, while production curtailment happens when a user adopts strategies to reduce their amount of produced energy.

2.1.2 Smart Grid actors

Within the smart grid scenario there are several important figures, some adapted from the old power grid environment, some completely new. In the following we will see what they are and what their role is [11, 12].

- **Prosumers.** The term prosumer defines an energy grid consumer who can also produce energy. However, in some contexts, this definition can be extended to grid users who do not necessarily produce energy, but can provide flexibility by manipulating their energy loads [11, 6, 12].
- **Aggregators.** An aggregator is an entity that mediates between prosumers and entities external to the grid, such as balance and network operators. It refers to a certain number of prosumers, usually in the same geographical location (in such case, they are called Local Flexibility Aggregators, or LFA [13]), allowing them to enter the energy market and to communicate with external entities. Also, it is responsible for the negotiations among prosumers and external entities.
- **Balance Responsible Party.** A *Balance Responsible Party* (BRP) is in charge of balancing energy supply and demand. It forecasts energy demand for prosumers and suppliers, and looks for the best solution in economical terms for maintaining the balance.
- **Distribution System Operator.** A *Distribution System Operator* (DSO) is responsible for the transfer of energy in a given region to and from end users through the medium and low voltage grid, and has to ensure the distribution system's long-term ability to meet electricity distribution demands.

2.2 Mathematical concepts

The work in this thesis is strongly based on mathematical construction and modelization. Consequently, some preliminary concepts and results in mathematics will be introduced in this section.

2.2.1 Known probability distributions

In Chapter 3 we will make use of some well-known probability distributions, and we will perform some operations on them. In this subsection, we will describe how these probability distributions are made, and how operations between them work.

A *Gamma* probability density function (PDF) is defined as

$$f(x) = \frac{x^{k-1} \cdot e^{-\frac{x}{\theta}}}{\theta^k \cdot \Gamma(k)} \tag{2.1}$$

where k and θ are two positive real numbers, named respectively *shape* and *scale*. The function Γ is defined on positive real numbers as

$$\Gamma(k) = \int_{t=0}^{\infty} t^{k-1} e^{-t} dt. \tag{2.2}$$

A *Beta* probability density function is defined as

$$f(x) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha,\beta)} \tag{2.3}$$

where α and β are two real positive numbers, called *shape parameters*. The function B is defined on real positive numbers as

$$B(\alpha,\beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)} \tag{2.4}$$

where Γ is the function defined in Eq. 2.2.

2.2.2 Operations on probability distributions

We will now describe how operations between PDFs are defined. Let f and g be the PDFs of two continuous, indipendent random variables. Then, the PDF of the sum of these variables can be calculated as

$$(f+g)(t) = \int_{-\infty}^{+\infty} f(x)g(t-x)dx \tag{2.5}$$

and the PDF of their product, as

$$(f \cdot g)(t) = \int_{-\infty}^{+\infty} f(x)g\left(\frac{t}{x}\right)\frac{1}{|x|}dx. \tag{2.6}$$

When instead of f or g we have a number N, the PDFs become:

$$(f+N)(t) = \int_{-\infty}^{+\infty} f(x)\delta_N(t-x)dx \tag{2.7}$$

$$(N \cdot f)(t) = \int_{-\infty}^{+\infty} f(x)\delta_N\left(\frac{t}{x}\right)\frac{1}{|x|}dx \tag{2.8}$$

where δ_N is the Dirac delta at N.

A very important property of these operations regards their expected value. Recall that, if a continuous random variable F has a function f as its PDF, the expected value of this variable can be calculated as

$$\mathbb{E}(F) = \int_{-\infty}^{+\infty} x \cdot f(x) dx. \tag{2.9}$$

It follows that:

- The expected value of $F+G$ is the sum of the expected value of F and the expected value of G.
- The expected value of FG is the product of the expected value of F and the expected value of G. This is only true if the two random variables are indipendent.

2.2.3 Game Theory

The core of game theory is the concept of *game*. A game (in normal form) is defined as a triplet $G = (U, S, Q)$, as follows.

- $U = \{U_1, \ldots, U_N\}$ is the set of players. They are the entities which influence the configuration of the game.
- $S = \{S_1, \ldots, S_N\}$. For any $i \in \{1, \ldots, N\}$, S_i is the set of player U_i's strategies. Given a player U_i, she[1] has the power to take decisions which will alter the configuration of the game; a strategy is one of these decisions.
- $Q = \{q_1, \ldots, q_N\}$. For any $i \in \{1, \ldots, N\}$, $q_i : \prod_{j=1}^{N} S_j \rightarrow \mathbf{R}$ is the payoff function for player U_i. They are functions which determine how much is the value of the current configuration of the game for each player. This value can be quantified by either something tangible like money, or abstract concepts like for example comfort.

All of the games that we will see in this thesis are *pure strategy* games: this means that, when a player is allowed to choose a strategy, the choice of that strategy is completely defined. However, there are also *mixed strategy* games where, when a player U_i is allowed to choose a strategy, for every strategy in S_i, she is assigned a certain probability to choose it.

Another fundamental concept in game theory is *Nash equilibrium*. A Nash equilibrium is a configuration of the game where every player, making the assumption that other players' strategies will not change, does not benefit by choosing any other strategy. In other words, this means that in a Nash Equilibrium every player is choosing the best possible strategy in response to other players' strategies. It is well-known that a mixed strategy game always admits the existence of a Nash equilibrium; however, this is not true for a pure strategy game, unless certain conditions are met, as we will see in Chapter 5.

[1] from now on in this thesis, this will be read as he/she

Chapter 3

Exploiting flexibility for demand-response: a use case on a grid in Cardiff.

3.1 Context

Since the last decades renewable energy is finding more and more place in our society and not only in mass production. Nowadays, in fact, normal users can produce energy, and devices such as photovoltaic panels are now affordable by families, so their use is rapidly increasing. This brings important changes to electricity grids scenarios: electricity does not necessarily travel one-way, and producers can decide to use their own energy instead of taking it from the grid. Moreover, they can *sell* what they produce, implying also important changes in the energy market. This has given birth to new figures in the energy distribution environment, and also electricity grids evolved into what we call *smart grids*.

The possibility of using smart grids gives many advantages for both users and electric energy providers [6, 7] in terms of costs, stability and environmental impact [5]. This chapter is focused on how the aforementioned energy market may be locally affected by this technology, how users and other external figures can maximize their revenues, how system operators can control peak loads, saving on upkeep and grid reinforcement costs and how these objectives can be pursued while minimizing discomfort for the users.

There have already been contributions on this aspect which will be shown within the Section 3.2. In our case we analyze a situation where tariffs are fixed, and smart grid users (in particular, prosumers [11, 6, 12]) have the possibility to produce electric energy[1] (e.g. by using photovoltaic systems), which, depending on their settings, may enable load *flexibility* to the energy market. The goal of this chapter is showing how to exploit such a flexibility to minimize costs and discomfort for grid users, and giving external parties, such as DSOs, the possibility to interact with them by buying flexibility and thus changing

[1] Definition of *prosumer* has been given in Chapter 2; in this chapter we will use the extended definition, referring as prosumers also those grid users who cannot produce energy, but can use flexible loads.

the total grid loads at a given time and at advantageous prices.

This work is focused on smart electricity grids. From now on, we will refer to the term *energy* for electric energy.

Three *levels* of smart grid that we focus on are defined in the following and heavily rely on the concept of *flexibility* [10], which we have defined in Chapter 2.

1. The first level is a scheduling problem: it leverages the flexibility to shift loads depending on the energy cost, PV production and user comfort.

2. The second level includes the possibility for different users to buy and sell energy.

3. In the third level, entities external to the grid, such as DSOs, offer a reward for users who shift their loads in order, for example, to avoid congestion at certain times: this is similar to the peak shaving problem.

Algorithms have been developed for each level and tested on data related to consumption, energy production, tariffs and smart settings in an existing grid in Cardiff, UK, and obtained from real data by applying some transformations in order to preserve anonymity. Moreover, data have been collected within the MAS^{2}TERING project[2], a three year technology-driven and business-focused project, aimed at developing an innovative information and communication technology platform for the monitoring and optimal management of local communities of prosumers. The obtained results have been collected at each level with our proposed approach and compared to the cost and comfort values obtained by the naive algorithm applied on those data in the context the MAS^{2}TERING project. The results indicate that our approach highly outperforms the naive algorithm.

In order to run tests for our work, we have fixed the following hypothesis:

- Tariff plans are pre-determined and each user may choose one among six possible tariff plans. We also show how our algorithm can be adapted to the case where real-time tariffs are not known in advance but can be forecasted. However, forecasting is out of the scope of this chapter.

- Consumption of grid users (without taking into account flexible loads) and photovoltaic (PV) generation [14, 15] can be considered known in advance. We also consider the case where they are rather represented by random variables.

- Flexibility intervals are fixed and depend on which flexibility plan is chosen out of nine options.

To sum up, the main contributions of the work of this chapter are:

- the algorithmic and mathematical formulation of the described concepts and levels in order to achieve the cost and discomfort optimization for each of them;

[2] `http://www.mas2tering.eu/`

- an optimization strategy which pursues cost and discomfort minimization by both using classic demand side management techniques, such as flexible loads allocations and peak shaving, and energy market strategies which imply economic transactions among prosumers within the same aggregator;
- the application of our methods on real data related to the energy grid of Cardiff, UK;
- the performances of our algorithms compared to the results obtained by a naive approach employed for the Cardiff energy data and the benefits we achieved with the proposed solution;
- a framework which takes into account the discomfort that prosumers may experience due to the load shifting or interrupting, with the goal of minimizing it and reducing their energy costs;
- a framework which takes into consideration uncertainty in load consumption, PV generation and user preferences.
- the employment of CUDA parallel programming on top of NVidia GPUs to show the reduced computational time of the parallel version of the brute-force algorithms in real scenarios.

3.2 Background

Literature on smart grid is quite extensive in different aspects.

At general level, there are environmental problems targeted for example by authors in [16] which analyze the impact of real-time pricing for electricity in terms of reduction for variance of demand, and consequent reduction of sulfur dioxide, nitrogen oxides and carbon dioxide emissions. Similarly, a survey published in 2016 [5] reported different works focused on environmental issues on smart grids, finding out a significant reduction in greenhouse gases emission, although results heavily depend on the presence of renewable energy sources.

At level of demand side management (DSM), there is a vast literature which covers interactions on many of the involved actors. One example is [17], which is a detailed descriptive work on the different types and aspects of DSM, and explains meticulously their nature, their possible applications, and results that can be obtained with their employment. The work in [14] is another survey, more focused on the types of tariffs and load forecasting. As far as the tariffs are concerned, various pricing schemes are described in detail, listing their advantages for load control. The forecasting is addressed indicating a number of statistical and AI-based methods showing how the latter achieve higher performances than the former thanks to their ability to handle the non-linearity of the data. One more work in [18] shows many of the aspects on which DSM problems may focus and how they are tackled, detailing cost and discomfort minimization in a similar fashion to our presented work, and proposing many techniques to solve these problems. For the scheduling problems, many works optimise with respect to a dynamic tariff. One example is represented by [19], which is conceptually similar to our level 1 optimization but takes into account the likeliness for a prosumer to pay more for a better time allocation, trying to forecast energy prices due to the dynamic tariff. Other similar works [20, 21, 15] consist of load commitment problems, although they put emphasis on the presence of plug-in electric vehicles (PEV) and take into account the PV generation. Other ways to tackle DSM problems are seen in [22, 23], which employ stochastic approaches. In particular [22] focuses on the economic aspects for the user and achieves relevant cost savings through load scheduling. The work in [23], focused on the optimal charging plans for PEV, shows advantages in both economic and peak demand aspects. Authors in [24] describe an approach related to another scheduling optimization problem which focuses more on users' preferences, and, after having forecasted prices in a Real Time Pricing (RTP) context, it allocates loads consequently using a shrinking horizon scheduling approach. This way, economic benefits for the users are obtained while load allocations remain close to user's preferences. Another different method is proposed in [25], where the Monte Carlo Search Tree method is applied for the demand side management in presence of RTP. DSM techniques are also used for power stability: for example, in [26] authors describe their usage for improving system security by using methods like load shifting and tariff incentives, and find out that applications of those may give significant results in terms of voltage stability and blackouts prevention. Several works also show scheduling problem strategies used in specific cases: for example, work in [27] considers the forecasting and load allocation problem regarding a water heater, whereas [28] shows an example of demand side management on a real solar house.

Level 2 optimization happens purely at economic side, as energy is not actually transferred between prosumers. A work in this domain is [29], which describes the state of art of actual peer-to-peer (P2P) energy trading, and the roles of distribution system operators (DSOs) and aggregators on this matter. Authors in [30] describe a similar scenario on a microgrid, where P2P energy transactions between prosumers are possible. In our level 2 optimization the aggregator plays a key role as it regulates the economic transactions between prosumers and manages their flexibility. On this matter, [9] describes an algorithm for an aggregator to gather flexibility from prosumers, making them actors in the energy market.

Our level 3 optimization problem is conceptually similar to the peak shaving problem, which is tackled in [31, 32] and it aims at reducing the total load of a grid in certain moments of the day. In fact, energy providers might encounter difficulties when the total load of the grid is high, therefore reducing loads is critical for them. Our considered level 3 optimization exploits the flexibility according to the offered reward and achieves a reduction of the grid's total load, effectively providing a way to perform peak shaving using flexibility. Works in [33, 34] are also related, where both the cost minimization and the peak avoiding problems are solved using a genetic algorithm.

When real time tariffs are present, level 1 and 3 optimizations are solved with a single approach. In fact, real time tariffs encourage users to shift their flexible loads when the grid's total load is not very high. This way, cost optimization (level 1) and peak shaving (level 3) are achieved by moving flexible loads in periods of the day when total load is lower. The reader notices that as real time tariffs are not given within our data, we considered the two levels separated.

In the following we will report some works that tackle level 1 and level 3 with single strategies. The work in [35] creates real time tariffs and their usage implies economical benefits to prosumers and a better total load control to system operators. Another similar work is [36] that uses RTP as an incentive to reduce maximum loads, and creates a multi-objective framework on this principle. Other works related to RTP take into account customers' behaviour: as an example, work in [37] uses a game theory approach to create a tariff. Regarding variable tariffs' effectiveness, in [38] there is a study on customers behavior with Critical Peak Pricing, showing an experiment involving 802 industrial and commercial users in Korea, where the effects on peak load reduction are shown. The work in [39] takes this concept further by examining customers' reactions on price-based and incentive-based demand response systems, and proposing a new model depending on their behaviour.

3.3 Description of the problem

Our problem considers three levels within the smart grid, which correspond to three use cases defined within the MAS^{2}TERING project and described in detail in [7].

Level 1

This level is local, and involves the single prosumer. The simple fact that there are flexible loads may offer a prosumer the possibility of saving money, for example by shifting the load to the periods of the day when energy is cheaper. Moreover, if the prosumers can also produce energy, it may be convenient to shift their loads when the production is at its peak, so that they can first consume their own energy. However, load shifting may be uncomfortable for the prosumers, since loads might be moved in time of the day not comfortable for them. In this case, the goal is to find the best possible flexible loads allocation in order to minimize costs and discomfort of the prosumer. In other words, this is a local *scheduling problem*.

Example: Let us suppose a prosumer with the following settings.

- An appliance scheduled at 16:00, with a flexibility interval between 8:00 and 20:00 and that operates for 1 hour.
- A tariff which becomes cheaper after 18:30.
- PV generation, which has a peak at 12:00 and exceeds the prosumer's consumption between 10:30 and 13:30.
- A preferred started time for the appliance, 14:00.

The goal is to minimize both the energy cost and the discomfort of the prosumer. As far as the cost is concerned, the tariff suggests as a good strategy to move the load after 18:30. However, as the prosumer produces more than what she consumes, it might be better to move the load between 10:30 and 13:30. This way, the prosumer may use her own energy instead of buying it from the market, and this solution is much more advantageous in economic terms. Regarding the user discomfort, the further the load is moved from the prosumer's preferred time (14.00), the less comfortable it would be for her.

Thus, the best time for minimizing costs is different than the best time for minimizing the discomfort. A good trade-off may be for example to start the load around 12:45; this way the user can still save money thanks to the energy production, and the load is not too far from her desired time. Depending on the priority that needs to be given to either the cost or to the discomfort components, the best solution may be different. For example, if the cost is the most important, the load may start at 12:30 in order to exploit more the PV generation. Otherwise, the load may begin at, let us say 13:00, so that it is closer to the user's desired time preference and for half an hour the prosumer can use the produced energy.

Level 2

This level regards prosumers linked by the same aggregator. After level 1 optimization, let us assume a prosumer A still produces an amount of energy higher than what she consumes at a certain time t of the day. Let us also consider another prosumer B which consumes energy at t. It might be convenient for B to buy energy from A and for A to sell her excess energy to B instead of interacting with the grid. The reason is that, in our specific case, no matter which tariff a prosumer has, the cost of the energy she buys from the supplier will always be higher than the cost of the energy she sells back to the grid, thus if A sells energy to B at an intermediate price between A's selling tariff and B's buying tariff, this can give a better economic return to both of them. In such a case, the challenge is to find which users are better to connect and which price maximizes the profit for both (buyer and seller). Also, prosumers may use flexibility to shift loads and obtain higher profits. Although the algorithm we will propose in this section takes into account flexibility, in our use case (Cardiff grid) this scenario does not exist as the overall production of energy is always lower than the overall consumption. Let us observe that, in such a scenario, transactions occur at economic level only through the coordination of an aggregator. The reader notices that the aggregator should be paid from the involved users for its crucial role. The amount and policies for its role are out of scope of this chapter and have been discussed in several deliverables within the exploitation activities of the MAS^{2}TERING project.

Example: In our context, daily time is divided in 96 units of 15 minutes each, which we will call Program Time Units (PTUs). Imagine that at PTU 49 (i.e. 12:00) prosumer B consumes 0.92 kWh, and prosumer A produces 1.45 kWh. Normally B would just buy energy from suppliers (for example at 0.24 £/kWh) and A would just sell it to the grid (suppose at 0.06 £/kWh). However, if B could obtain energy directly from A at an intermediate price, e.g. 0.15 £/kWh (average between 0.06 £/kWh and 0.24 £/kWh), it would be better for both of them in economical terms - so the best outcome would be with *A* selling 0.92 kWh to *B*. After doing that, if there is a prosumer *C* which buys 1.5 kWh at a price of 0.2 £/kWh, *A* could sell to *C* its remaining 0.53 kWh at an intermediate price of 0.13 £/kWh (average between 0.06 £/kWh and 0.2 £/kWh): this way each prosumer (*A*, *B* and *C*) saves money.

Level 3

In this level, external actors such as BRPs and DSOs are involved. Here, one of them may have specific needs for shaving loads or reducing them at a specific time of the day, and the aggregator can provide enough flexibility to fulfill their needs. If this happens, the external actor may be willing to pay for the flexibility, and with that income the aggregator can offer a reward to the prosumer who provides that flexibility. Of course, from the prosumer's point of view, the reward should be high enough to compensate the load shifting to a non-optimal configuration in economic terms and the additional generated discomfort. In this level, the optimization generates the best load profiles for the prosumers, and the best offer for the external actor. As discussed in Section 3.2, prosumers may employ real-time tariffs and therefore, in such a case, there is no distinction between Level 1 and Level 3 optimization, since tariffs have been planned in order to encourage load shifting according to the third party's policies.

Example: If at a certain PTU the DSO needs the total load to be lowered, flexibility may be offered to reduce the load by a certain amount. For example, if the DSO wants to reduce the total load at 12:00, there may be a prosumer who can move a flexible load, for example, from 12:00 to 15:00. This will have a cost for the prosumer as loads are moved from their optimal allocation; in this case, our prosumer would have its energy cost increased by a certain amount, say 0.08 £, after this shifting. Furthermore, the new allocation may be less comfortable to the prosumer, who might have preferred her load to start at 12:00. The discomfort thus generated can be balanced with a certain amount of money that will be calculated through defined policies or strategies. In our example, let us say the discomfort can be quantified as 0.15£. Therefore, the DSO is likely to offer a reward to the prosumer for the flexibility offered, which in this case may be $0.08+0.15$ £, so the prosumer is refunded for both the increased cost and the potential discomfort.

Explanation of the levels hierarchy

In literature, the optimization levels mentioned above have been treated quite extensively, and not always separately. It is indeed quite common to find works that tackle the problems in an aggregate manner, as shown in Section 3.2. The reason is that, usually, RTP tariffs are used as a mean to achieve peak regulations by rewarding prosumers that contributed: thus peak shaving (level 3) and user cost optimization (level 1) are achieved at the same time.

In our case, we have decided to run level 1 optimization before the other two, for four main reasons:

- If level 1 is performed before level 2, the excess energy produced by the prosumer is employed for self-consumption instead of being transferred. This reduces the number of needed energy transactions, and is efficient for the prosumer in economic terms, since self-consumption is more advantageous than energy trading.

- After level 1 optimization, it is still possible to perform the other two levels. If level 3 was performed first, then it would be impossible to perform level 1 optimization. This is due because after level 3, flexibility usage is agreed with a third party, and therefore it is not possible to change it anymore.

- Level 1 optimization allows to quantify the best possible gain of each prosumer in terms of cost and comfort. This gives an advantage for the prosumers during the negotiation with the aggregator. Furthermore, it is also possible to estimate how much the potential discomfort is worth in economic terms. This allows calculating a suitable reward for each prosumer.

- In terms of cost, for prosumers it is usually cheaper to use their produced energy, rather than selling it to the grid or to other prosumers. This implies that the highest savings in terms of costs are obtained by giving priority to level 1 over level 2.

The proposed scheme is not a joint optimization, but is divided into levels; however this subdivision does not decrease the efficiency of the overall optimization. The reason why savings are optimal when performing level 1 before level 2 is the following. Suppose a prosumer has a flexible load at a certain PTU t which consumes a certain amount of energy, and that the same prosumer produces the same amount of energy at other PTUs. On the one hand, if the prosumer decides to sell her excess energy, the chosen tariff will not compensate the cost of the load, since selling tariffs are lower than buying tariffs. On the other hand, shifting the load allows the prosumer to consume the produced energy, with no cost for the load. Furthermore, performing the first two levels before level 3 guarantees to the prosumers the highest possible return in economic and comfort terms. This occurs because prosumers earn from level 1 and level 2, while from level 3 they only get compensation for the loss of money or comfort caused by their agreement with the external party. Therefore, if the economic/comfort position of the prosumers before performing level 3 is profitable, the reward that they will obtain from that level will be higher. Even in the case of a joint optimization (i.e. performing all the levels at the same time), the profit obtained by the agreement with the external party is determined by how the prosumers are using flexibility at the moment of the agreement. Since levels 1 and 2 give prosumers the most profitable flexibility profiles, they will be guaranteed the highest possible return from the agreement.

Level 2 can still exploit flexibility and, therefore, in the general case it has to be performed before level 3 (which assumes that flexible loads are fixed). This is due to the fact that in level 3 an external actor buys flexibility from the aggregator (indirectly, from the prosumer), and, after that flexibility is agreed with the external party, flexible load allocation cannot be changed anymore. This is also another reason for which level 1 optimization has to be performed before level 3. Only when level 2 does not need to employ flexibility, it is not important which one is run first (level 2 or level 3).

3.4 Data Sets

In this section we describe the data set we have used to conduct our simulations. Reference data about a grid in Cardiff, UK, have been collected, diversified and projected in future smart grid scenarios [40, 41] with different settings. In the following we will report details of such data:

- Daily energy consumption of 184 grid users for the following five types of settings:

  - *High*, *medium* and *low* smart technology use (3 options);
  - 2020 and 2030 settings (2 options);
  - *Business As Usual* (*BAU*) and *Green* settings (2 options);
  - Day the week (*Weekdays*, *Saturdays* and *Sundays*) (3 options);
  - Season (*Autumn*, *Winter*, *Spring*, *Summer* and *High Summer*) (5 options);

  We call a *scenario* a combination of the above settings, obtained by choosing one option for each of the five types of settings. Moreover, consumption is recorded for each of the 96 PTUs (each PTU refers to a 15 minutes interval). The total number of possible combinations of these settings, and therefore of possible scenarios, is 180. Hence, each grid user has a consumption vector for each of the 180 possible scenarios and each vector consists of 96 consumption values. Table 3.1 shows the number of prosumers with flexible loads, number of total flexible loads and number of prosumers which can produce energy. These values may change depending on the settings: scenarios with the 2030, such as *Green* or *high* settings, will have more flexible loads (**#loads**), and prosumers with them (**#pros.**), than scenarios with the 2020 (*BAU* or *low* settings). The first row of Table 3.1 (2030 - *Green* - *high*) is the one with the highest values, in which 166 users out of 184 have at least one flexible load, and 41 users out of 184 have a PV system. The 2020/2030 and *Green*/*BAU* settings have been introduced in a report from UK National grid [41], and describe how production/consumption would be if certain policies are adopted for the grid. The *high*/*medium*/*low* settings indicate the grade of penetration of some smart technologies inside the defined grid scenario.

- Descriptions of energy consumption for cycles of various appliances in different period of the week (*Weekdays*, *Saturdays*, *Sundays*) and number of appliances in each house and scenario. Available appliances are the following:

Settings	**#pros.**	**#loads**	**PV users**
2030 Green high	166	380	41
2030 BAU high	92	152	20
2020 Green high	101	190	25
2020 BAU high	55	76	15
2030 Green medium	101	190	41
2030 BAU medium	43	58	20
2020 Green medium	55	76	25
2020 BAU medium	29	38	15
2030 Green low	56	76	41
2030 BAU low	25	30	20
2020 Green low	29	38	25
2020 BAU low	10	10	15

Table 3.1: Number of prosumers per setting. The columns describe, respectively, settings, number of prosumers with flexible loads, number of total flexible loads and number of prosumers which can produce energy.

  - Dishwasher
  - Electric Oven
  - Electric Vehicle
  - Freezer
  - Fridge
  - Tumble dryer
  - Washing machine
  - Heat pump

- Energy production data for each prosumer with PV systems.
- User tariffs data, for selling and buying energy. The former includes only one tariff shared by all the users, whereas the latter consists of 6 available tariffs, which depend on the the users' choices.
- Flexibility settings for each prosumer: times of the day where flexible loads may be allocated for each smart appliance. They are pre-determined, but depend on the habits and choices of the prosumers. According to these habits, prosumers are assigned different profiles: some examples of these profiles may be *worker*, *not worker* or *always at home*. In our case there are 9 pre-determined profiles, which identify 9 different possibilities for flexibility settings. One of the 9 combinations can be seen in Table 3.2 whereas the entire list is publicly available[3].

[3] https://github.com/flilliu/UASmartGrid

The DSO of the aforementioned grid provided general data such as total year consumption, number and typologies of users, and average values of energy consumption for every user typology and period of the week/year. The data set we worked on has been anonymized. The initial load allocation present in these data has been performed using a naive algorithm which will be considered as the baseline we compare our results against.

Appliance	**ST**	**FT**	**ST**	**FT**
Washing Machine	09:00	14:00	17:00	22:00
Tumble Dryer	09:00	22:00		
Dishwasher	13:00	17:00	20:00	00:00
Electric Oven	12:00	15:00	19:00	22:30
Electric Vehicle	00:00	05:00		

Table 3.2: One of the 9 possibilities for flexibility settings. For each appliance, start time (ST) and finish time (FT) of each flexibility interval is indicated, and in some cases there can be more than one interval, like washing machines, dishwashers and electric oven.

A list of all the details on the data and the underlying grid, named *Bologna*, can be found in some of the produced deliverables of the MAS^{2}TERING project[4] [11, 42, 7].

3.5 Methods and algorithms

In this section we discuss in details the algorithms we have introduced. Sections 3.5.1 and 3.5.2 describe, respectively, the algorithm of level 1 optimization when variables such as power consumption, PV generation and user preferences (detailed in Section 3.5.1) are known in advance, and its version which takes into account uncertainty. Section 3.5.3 describes the level 2 optimization algorithm whereas Section 3.5.4 shows a different version of the same algorithm which exploits parallel computation. Section 3.5.5 describes the level 3 optimization algorithm, that requires several external inputs. For this reason, an automatic version of it that looks for the highest possible peak reduction has been developed as well, described in Section 3.5.6, and used for our experiments in Section 3.6.3.

3.5.1 Level 1 Optimization - Algorithm without Uncertainty

We recall that the level 1 is where prosumer's flexible loads are moved so that energy costs and user discomfort are minimized, without interacting with external actors or other grid users. Let us choose a prosumer and a scenario, i.e. season, time of the week and smart technology use. Let Z be the number of PTUs the day consists of: in our case, $Z = 96$. Then the following data represent the input of the algorithm for the chosen two elements.

[4]To obtain the data please send an email to the MAS^{2}TERING coordinator

Input of the algorithm

- **Daily net energy consumption of the prosumer**. This is a vector **N** of length Z, where the t-th component contains the net energy consumption at PTU t, which corresponds to the difference between consumed and produced energy at t. It represents the energy consumed by interruptible loads (and not by shiftable loads).

- **Shiftable loads**. These are vectors of length Z (we will refer to those as $\mathbf{v_j}$, where j goes from 1 to m, the number of smart appliances of the prosumer), where the t-th component contains energy consumption at PTU t. There is a shiftable load for each smart appliance of the prosumer. For example, fridges and freezers are always active, but their consumption is not uniform and therefore their energy loads may be shifted.

- **Interruptible loads**. These are vectors of length Z, where the t-th component contains energy consumption at PTU t. We will refer to those as $\mathbf{w_j}$, where j goes from 1 to b, the number of interruptible appliances of the prosumer. There is an interruptible load for each device of the prosumer whose operation can be interrupted and resumed, such as heat pumps. Moreover, the loads of these devices at a specific time can be changed, but the total daily energy load must remain the same.

- **Tariffs**. These are two positive vectors of length Z. The first, **T**, contains the tariff at which the prosumer buys energy from the grid at each PTU. The second, **S**, contains the tariff at which the prosumer sells energy to the grid at each PTU.

- **Maximum load**. This is a number M, which determines the maximum energy consumption the prosumer may have at any PTU.

- **Flexibility settings**. This determines how each shiftable load of the prosumer can be allocated.

- **User preferences**. For each shiftable load, the most comfortable allocations for the prosumer are known. Furthermore, for energy consumption relative to interruptible loads, prosumers' preferences are known too.

A shiftable load can be moved in time: in mathematical terms, *moving a shiftable load* $\mathbf{v_j}$ means rotating the components of the vector $\mathbf{v_j}$ by a certain number of positions. $P_k(\mathbf{v_j})$ will denote the vector obtained by rotating each component of $\mathbf{v_j}$ by k positions forward. Given an interruptible load $\mathbf{w_i}$, its t-component can become zero (which means interrupting the device during the PTU t) as long as the overall sum of all the components of $\mathbf{w_i}$ does not change.

There might be some slots indicated within the flexibility settings at which a shiftable load cannot be moved. The set of the allowed slots will be called I_j.

Objective, cost, discomfort functions

The objective of this optimization level is to minimize costs and discomfort of the prosumer. In order to achieve this, we define two functions C and D, which describe, respectively, the cost of the electricity for the prosumer and her discomfort. The discomfort will be weighted using a parameter $\alpha \geq 0$. Therefore, the objective is to minimize the *goal function*, which has been defined in [18] and that we introduce as:

$$G = C + \alpha \cdot D. \tag{3.1}$$

In particular, for $\alpha = 0$ the problem becomes a cost minimization task.

We will now describe the cost and discomfort functions.

Let $\mathbf{L}$ be an energy consumption vector. Let us define $\mathbf{L}^+$ as $\mathbf{L}$ with each negative component changed to zero, and $\mathbf{L}^-$ as $\mathbf{L}$ with each positive component changed to zero. A value is positive (negative) when the consumed energy is higher (lower) than the produced energy. The energy cost will be calculated as:

$$C(\mathbf{L}) = \mathbf{L}^+ \cdot \mathbf{T} + \mathbf{L}^- \cdot \mathbf{S} \tag{3.2}$$

User discomfort highly depends on each prosumer and cannot be described with an exact formula. Several works in literature tackled the problem. The discomfort concerning the shiftable loads is generated by the load shifting: a prosumer chooses when to activate a certain device, and the discomfort increases by moving further and further away the activation time. We have employed the same approach described in [18, 43]. In particular, we adopted a variation of the formula suggested in [43]. Let us suppose $\mathbf{v_i}$ is a shiftable load, pr_i the PTU when the prosumer would like to begin the load, and ac_i the PTU when the load actually starts. The discomfort relative to the load is then:

$$D_{shift}(\mathbf{v_i}) = \rho \cdot |pr_i - ac_i|^k \tag{3.3}$$

where ρ and $k \geq 1$ are two real numbers, and ρ acts as a factor of conversion in terms of money. The total discomfort generated by the shiftable loads is then equal to

$$D_{shift} = \sum_{i=1}^{m} D_{shift}(\mathbf{v_i}). \tag{3.4}$$

The discomfort concerning the interruptible loads (which mostly involve temperature related devices) depends on the difference between the generated and desired temperature. We have employed the same approach described in [44, 45].

The formula we decided to adopt is an adaptation of that defined in [45]. For the load $\mathbf{w_j}$, let $\mathbf{tw_j}$ be the vector of the temperatures desired by the prosumer at each time, and $\mathbf{ta_j}$ the vector of the temperatures actually provided at each time. More in detail, let us fix a time unit t, $\mathbf{tw_j}[t]$ is the temperature the prosumer desires at time t, and $\mathbf{ta_j}[t]$ the provided temperature. The discomfort function is then defined as:

$$D_{int}(\mathbf{w_j}) = \mu \cdot \sum_{t=1}^{Z} |\mathbf{tw_j}[t] - \mathbf{ta_j}[t]|^h$$

where, as seen in the previous case, μ and $h \geq 1$ are real numbers. As suggested in [45], we set $h = 2$. It has to be noted that our data regard energy consumption and not temperature; however temperature at a certain PTU depends on the quantity of used energy. For this reason, calling $\mathbf{w_j}[t]$ the $t-$th component of the vector $\mathbf{w_j}$ and $\mathbf{cw_j}$ the vector describing the consumption desired by the prosumer, the discomfort function we actually used is:

$$D_{int}(\mathbf{w_j}) = \mu \cdot \sum_{t=1}^{Z} \left| \sum_{u=1}^{t} \mathbf{w_j}[u] - \sum_{u=1}^{t} \mathbf{cw_j}[u] \right|^h \tag{3.5}$$

The total discomfort function for interruptible loads is then

$$D_{int} = \sum_{j=1}^{b} D_{int}(\mathbf{w_j}).$$

Therefore, if we combine what we have defined so far, the total discomfort function is defined as:

$$D = D_{shift} + D_{int}.$$

Pseudocode of the Algorithm

The optimization works as it follows:

1. For each shiftable load $\mathbf{v_j}$, a permitted allocation is chosen. In other words, a number $k_j \in I_j$ is chosen for each j.

2. Each $\mathbf{v_j}$ thus allocated is added to the net energy consumption vector. That is, the following vector is created:

$$\mathbf{L}_{k_1,\dots,k_m} = \mathbf{N} + P_{k_1}(\mathbf{v_1}) + \ldots + P_{k_m}(\mathbf{v_m})$$

3. For each combination of the k_i, interruptible loads are modified depending on the PTUs where tariff changes. Since the vectors $\mathbf{w_j}$ are already considered in $\mathbf{N}$, any modification to them also changes the consumption vector $\mathbf{L_{k_1,\dots,k_m}}$. To be more specific on how the $\mathbf{w_j}$ are changed, let us consider the tariff vector $\mathbf{T}$ at time t.

   - If $\mathbf{T}[t] = \mathbf{T}[t+1]$, nothing changes.
   - If $\mathbf{T}[t] > \mathbf{T}[t+1]$, the value $\mathbf{w_j}[t]$ is reduced by a certain quantity Q_j, and the value $\mathbf{w_j}[t+1]$ is increased by the same quantity Q_j.
   - If $\mathbf{T}[t] < \mathbf{T}[t+1]$, the value $\mathbf{w_j}[t]$ is increased by a certain quantity Q_j, and the value $\mathbf{w_j}[t+1]$ is reduced by the same quantity Q_j.

   The quantity Q_j is determined as follows. Let Q_j^1 be the maximum quantity which can be moved without unsatisfying the constraints of not exceeding the device's maximum energy output and the load limit for that PTU. Let Q_j^2 be the optimal value for minimizing the combined value of the cost and the discomfort function for the interruptible load: it can be calculated by using derivatives, since this is a convex function. More precisely, the part of this function relative to the interruptible load is

$$G(x) = (|\mathbf{T}[t+1] - \mathbf{T}[t]|) \cdot x + \mu \cdot x^h$$

The minimum of such a function will be achieved when the derivative is equal to zero. Therefore, the explicit formula for Q_j^2 becomes

$$Q_j^2 = \left(\frac{|\mathbf{T}[t+1] - \mathbf{T}[t]|}{\mu \cdot h} \right)^{\frac{1}{h-1}} \tag{3.6}$$

Then, Q_j is chosen as the minimum between Q_j^1 and Q_j^2.

4. Interruptible loads can be changed depending on $\mathbf{L}$'s values. More in detail, let us consider the $\mathbf{L}$ vector at PTU t.

   - If $\mathbf{L}[t]$ and $\mathbf{L}[t+1]$ have the same sign, nothing happens.
   - If $\mathbf{L}[t] > 0$ and $\mathbf{L}[t+1] < 0$, the value $\mathbf{w_j}[t]$ is decreased by a certain quantity Q_j, and the value $\mathbf{w_j}[t+1]$ is increased by Q_j.
   - If $\mathbf{L}[t] < 0$ and $\mathbf{L}[t+1] > 0$, the value $\mathbf{w_j}[t]$ is increased by a certain quantity Q_j, and the value $\mathbf{w_j}[t+1]$ is reduced by Q_j.

   The quantity Q_j is calculated similarly to step 3.

5. The goal function relative to the total load with the chosen allocations is calculated. We are calculating $G(\mathbf{L}_{k_1,\ldots,k_m})$, by combining Eq. 3.1, 3.2 and 3.4.

6. The first five steps are repeated for every possible combination of $k_j \in I_j$. The number of times this process is repeated is then $\prod_{j=1}^{m} |I_j|$. The results depend only on the choice of the k_j made at step 1: for this reason, the steps described so far can be run in parallel for every possible choice of the indexes k_j.

7. The allocations which achieved the lowest value for the goal function are chosen: in other words, the $k_1, \ldots, k_m$ which minimize the goal function $G(\mathbf{L}_{k_1,\ldots,k_m})$.

8. A check is performed in order to ensure that energy consumption with the chosen allocation does not exceed the maximum load limit at any PTU. That is, the inequality

$$\mathbf{L}_{k_1,\ldots,k_m}[t] \leq M \tag{3.7}$$

   has to hold for each $t \in \{1, \ldots, Z\}$. If this happens, k_j are defined as optimal, and the resulting consumption vector $\mathbf{L}_{k_1,\ldots,k_m}$ will be denoted by $\mathbf{L}$. Otherwise, this configuration is discarded and the algorithm goes back to step 7, taking the second best choice of the k_j.

9. The output of the algorithm are the optimal values for the k_j, and the values of Q_j.

Example

Let us assume we have two appliances with shiftable loads and one with interruptible loads. Moreover, let us assume 7 PTUs with the following parameters:

$$\begin{array}{rl} \mathbf{N} = & [1.0\ \ 0.3\ \ -0.1\ \ -0.7\ \ -0.3\ \ 0.7\ \ 1.1] \\ \mathbf{v_1} = & [0.2\ \ 0.3\ \ 0.4\ \ 0.2\ \ 0.5\ \ 0.0\ \ 0.0] \\ \mathbf{v_2} = & [0.1\ \ 0.4\ \ 0.2\ \ 0.1\ \ 0.5\ \ 0.0\ \ 0.0] \\ \mathbf{w_1} = & [0.3\ \ 0.0\ \ 0.0\ \ 0.0\ \ 0.0\ \ 0.0\ \ 0.0] \\ \mathbf{T} = & [1.2\ \ 0.8\ \ 0.8\ \ 0.6\ \ 0.8\ \ 1.2\ \ 1.0] \\ \mathbf{S} = & [0.3\ \ 0.3\ \ 0.3\ \ 0.3\ \ 0.3\ \ 0.3\ \ 0.3] \\ M = & 1.5 \end{array} \tag{3.8}$$

Where consumption data are expressed in kWh and tariffs in £/kWh. We recall that $\mathbf{w_1}$ is already counted in $\mathbf{N}$.

Also, let us suppose the parameters for discomfort are:

$$\begin{array}{ll} pr_1 & = 2 \\ pr_2 & = 1 \\ \alpha & = 0.25 \\ \rho & = 0.25 \\ k & = 1.5 \\ \mu & = 0.3 \\ h & = 2 \end{array} \tag{3.9}$$

In this case, pr_1, pr_2, α, k and h are numbers, while ρ and μ are a factor of conversion between discomfort and money, and are therefore expressed, respectively, in £and £/$(kWh)^t$, according to their respective formulas.

Suppose that flexibility settings allow us shifting $\mathbf{v_1}$ and $\mathbf{v_2}$ by 0, 1 or 2 places forward. Thus, for example, only these three configurations for $\mathbf{v_1}$ are allowed:

$$\begin{array}{rl} P_0(\mathbf{v_1}) = & [0.2\ \ 0.3\ \ 0.4\ \ 0.2\ \ 0.5\ \ 0.0\ \ 0.0] \\ P_1(\mathbf{v_1}) = & [0.0\ \ 0.2\ \ 0.3\ \ 0.4\ \ 0.2\ \ 0.5\ \ 0.0] \\ P_2(\mathbf{v_1}) = & [0.0\ \ 0.0\ \ 0.2\ \ 0.3\ \ 0.4\ \ 0.2\ \ 0.5] \end{array}$$

So, since $|I_1| = |I_2| = 3$, there are 9 possibilities to explore.

The procedure starts, calculating the $\mathbf{L}_{k_1,k_2}$ vectors:

$$\begin{array}{rl} \mathbf{L}_{0,0} = & [1.3\ \ 1.0\ \ 0.5\ \ -0.4\ \ 0.7\ \ 0.7\ \ 1.1] \\ \mathbf{L}_{0,1} = & [1.2\ \ 0.7\ \ 0.7\ \ -0.3\ \ 0.3\ \ 1.2\ \ 1.1] \\ \mathbf{L}_{1,0} = & [1.1\ \ 0.9\ \ 0.4\ \ -0.2\ \ 0.4\ \ 1.2\ \ 1.1] \\ \mathbf{L}_{1,1} = & [1.0\ \ 0.6\ \ 0.6\ \ -0.1\ \ 0.0\ \ 1.7\ \ 1.1] \\ \mathbf{L}_{2,0} = & [1.1\ \ 0.7\ \ 0.3\ \ -0.3\ \ 0.6\ \ 0.9\ \ 1.6] \\ \mathbf{L}_{2,1} = & [1.0\ \ 0.4\ \ 0.5\ \ -0.2\ \ 0.2\ \ 1.4\ \ 1.6] \\ \mathbf{L}_{0,2} = & [1.2\ \ 0.6\ \ 0.4\ \ -0.1\ \ 0.4\ \ 0.8\ \ 1.6] \\ \mathbf{L}_{1,2} = & [1.0\ \ 0.5\ \ 0.3\ \ 0.1\ \ 0.1\ \ 1.3\ \ 1.6] \\ \mathbf{L}_{2,2} = & [1.0\ \ 0.3\ \ 0.2\ \ 0.0\ \ 0.3\ \ 1.1\ \ 2.1] \end{array} \tag{3.10}$$

Since $\mathbf{L}_{1,1}$'s sixth component exceeds M, this case is discarded. Also, each of the five remaining vectors (that is, $\mathbf{L}_{2,0}$, $\mathbf{L}_{0,2}$, $\mathbf{L}_{2,1}$, $\mathbf{L}_{1,2}$, $\mathbf{L}_{2,2}$) has the seventh component exceeding M, so they will not be considered.

The cost function is calculated on these vectors. For example, for $\mathbf{L}_{0,0}$, it results:

$$\begin{aligned} \mathbf{L}_{0,0}^{+} &= [1.3 \quad 1.0 \quad 0.5 \quad 0.0 \quad 0.7 \quad 0.7 \quad 1.1] \\ \mathbf{L}_{0,0}^{-} &= [0.0 \quad 0.0 \quad 0.0 \quad -0.4 \quad 0.0 \quad 0.0 \quad 0.0] \end{aligned}$$

The cost function will be therefore:

$$C(\mathbf{L}_{0,0}) = \mathbf{L}_{0,0}^{+} \cdot \mathbf{T} + \mathbf{L}_{0,0}^{-} \cdot \mathbf{S} = 5.26 - 0.12 = 5.14\pounds$$

Using the same procedure, the cost in the other two cases is calculated, resulting $C(\mathbf{L}_{0,1}) = 5.25$ £ and $C(\mathbf{L}_{1,0}) = 5.16$ £.

Now, let us calculate the discomfort. Regarding shiftable loads, the formula from Eq.(5.3) becomes:

$$D_{shift}(\mathbf{v_i}) = 0.25\pounds \cdot |pr_i - ac_i|^{1.5}.$$

Applying on the valid configurations, it becomes:

$$\begin{aligned} D_{shift}(\mathbf{v_1}) &= 0.25\pounds \cdot |2-0|^{1.5} \approx 0.71\pounds \quad \text{for } k_1 = 0 \\ D_{shift}(\mathbf{v_1}) &= 0.25\pounds \cdot |2-1|^{1.5} = 0.25\pounds \quad \text{for } k_1 = 1 \\ D_{shift}(\mathbf{v_2}) &= 0.25\pounds \cdot |1-0|^{1.5} = 0.25\pounds \quad \text{for } k_2 = 0 \\ D_{shift}(\mathbf{v_2}) &= 0.25\pounds \cdot |1-1|^{1.5} = 0\pounds \quad \text{for } k_2 = 1 \end{aligned}$$

For the loads listed above, their discomfort becomes:

$$\begin{aligned} D_{shift}(\mathbf{L}_{0,0}) &= 0.96\pounds \\ D_{shift}(\mathbf{L}_{0,1}) &= 0.71\pounds \\ D_{shift}(\mathbf{L}_{1,0}) &= 0.50\pounds \end{aligned}$$

The only thing left is Q_1. The only interruptible load is at PTU 1, and is 0.3 kWh. Since the tariff $\mathbf{T}$ at PTU 2 is lower than that at PTU 1, this load can be interrupted. Now, $Q_1^1 = 0.3$ kWh since at PTU 2 the power constraint would still hold. As for Q_1^2, applying Eq. (5.6), it can be calculated as it follows:

$$Q_1^2 = \left(\frac{|0.8\pounds/kWh - 1.2\pounds/kWh|}{0.3\pounds/(kWh)^2} \right)^{\frac{1}{2-1}} \approx 1.33kWh$$

Thus, since $Q_1^1 < Q_1^2$, Q_1 will be 0.3 kWh.

The Q_1 calculated this way holds for each of the $\mathbf{L}_{0,0}$, $\mathbf{L}_{0,1}$ and $\mathbf{L}_{1,0}$. The associated discomfort value is obtained using formula (5.5) $D_{int} = 0.3\pounds/(kWh)^2 \cdot (0.3kWh)^2 \approx 0.03\pounds$.

The goal function can now be calculated. We have:

$$
\begin{aligned}
&G(\mathbf{L}_{0,0}) = C(\mathbf{L}_{0,0}) + \alpha \cdot (D_{shift}(\mathbf{L}_{0,0}) + D_{shift}(\mathbf{L}_{0,0})) \\
&= 5.14£ + 0.25 \cdot (0.96£ + 0.03£) \approx 5.39£ \\
&G(\mathbf{L}_{0,1}) = C(\mathbf{L}_{0,1}) + \alpha \cdot (D_{shift}(\mathbf{L}_{0,1}) + D_{shift}(\mathbf{L}_{0,1})) \\
&= 5.25£ + 0.25 \cdot (0.71£ + 0.03£) \approx 5.44£ \\
&G(\mathbf{L}_{1,0}) = C(\mathbf{L}_{1,0}) + \alpha \cdot (D_{shift}(\mathbf{L}_{1,0}) + D_{shift}(\mathbf{L}_{1,0})) \\
&= 5.16£ + 0.25 \cdot (0.50£ + 0.03£) \approx 5.30£
\end{aligned}
$$

$k_1 = 1$ and $k_2 = 0$ represent the optimal configuration, and the output value for Q_1 is 0.3 kWh.

3.5.2 Level 1 Optimization - Algorithm with Uncertainty

The algorithm shown in 3.5.1 works in an environment where the input is known in advance. However, in many cases, data is present with a certain level of uncertainty that needs to be considered. For this reason, another version of the algorithm has been developed, taking into account such an aspect.

Data with Uncertainty

When uncertainty is present, forecasting can be performed with a certain degree of precision. However, forecasting is out of the scope of this work. In our work, we employ some models present in literature to tackle uncertainty on the following factors.

- **Energy consumption**. The energy consumption of a prosumer can be predicted from her historical consumption data and her behavior, or via more complex methods. When data is not enough to perform an efficient forecasting, energy consumption can however still be modelized as a probability density function (PDF) of a random variable. Some works attempted to do this, and the most efficient model has been found as a Gamma random variable [46] (see Eq. 2.1), which is what we have used in our work.

- **PV generation**. There have been several attempts to create a probability model for electricity generation from PV panels. These had to take into account various factors, such as season and weather. The work in [8] takes into account many of these factors in order to create a model for solar irradiation, and another work [47] concluded that PV generation is best modelized with a Beta random variable PDF (see Eq. 2.3). We have therefore used the work in [47] for our model.

- **User preferences**. The work in [48] makes an attempt to predict user preferences and the general algorithm might employ that schema. In our case users preferences can be deduced from the available data, so creating a different model has not been necessary. In fact, it is enough to compare the average consumption data of a given prosumer with the consumption of prosumers without flexible loads to easily detect the PTUs with values higher than the average: those will correspond to the prosumer's favorite times for flexible loads.

Tariffs (RTP) might also not be given and methods to forecast them have been proposed in literature [49]. In the specific case of the Cardiff smart grid, tariffs only depend on the hour and the prosumer's price plans: in other words, they are Time of Use (ToU) tariffs, so RTP does not apply here. However, RTP tariffs can still be integrated in the algorithm without much trouble, as it will be shown later.

Input of the Algorithm with Uncertainty

The input, output and functions that will be used for this algorithm are conceptually the same of the algorithm defined in Section 3.5.1. The objective is also the same: find the allocations of shiftable loads $\mathbf{v_j}$ which are compatible with the prosumer's flexibility settings and minimize a combination of energy cost and discomfort for the user.

The input for the algorithm is constituted by the following variables.

- **Daily net energy consumption of the prosumer**. The concept is the same of Section 3.5.1, but in this case uncertainty of load consumption is considered. For this reason, this variable is represented by a vector $\overline{\mathbf{N}}$ of length Z, where each of its components corresponds to the PDF of a random variable. Net energy consumption at a certain PTU is defined as the difference between energy consumed and energy produced at that time. For this reason, each component of $\overline{\mathbf{N}}$ is obtained as difference between two PDFs with distributions Gamma and Beta, respectively.

- **Shiftable loads**. As described in Section 3.5.1, they are vectors $\mathbf{v_j}$ of length Z, with j ranging from 1 to m.

- **Interruptible loads**. As described in Section 3.5.1, they are vectors $\mathbf{w_j}$ of length Z, with j ranging from 1 to b.

- **Tariffs**. Similarly to Section 3.5.1, since in the Cardiff grid the tariffs are ToU, they are described as a couple of vectors $\mathbf{T}$ and $\mathbf{S}$ of length Z. In case of RTP tariffs, however, they can be instead considered as two vectors $\overline{\mathbf{T}}$ and $\overline{\mathbf{S}}$ of length Z: in each of them, every component is a PDF of a random variable which describes the probability distribution of tariffs (for buying and selling energy, respectively) at the chosen PTU.

- **Maximum load**. As described in Section 3.5.1, it is a number M.

- **Flexibility settings**. Same as described in Section 3.5.1. In particular, for each shiftable load $\mathbf{v_j}$, flexibility settings are represented by the set I_j.

- **User preferences**. In this case, this is represented by some vectors $\mathbf{p_j}$, one for each shiftable load and each one with length Z. Each component $\mathbf{p_j}[t]$ represents the probability for the prosumer to begin the load $\mathbf{v_j}$ at PTU $\mathbf{s_j}[t]$. For this reason, the constraints

$$\sum_{t=1}^{Z} \mathbf{p_j}[t] = 1$$

$$\mathbf{p_j}[t] = 0 \;\; \forall t \notin I_j$$

hold for each $\mathbf{p_j}$. In particular, if the preference of the user for a load $\mathbf{v_j}$ is known to be at a specific time t_0, the vector $\mathbf{p_j}$ becomes:

$$\mathbf{p_j}[t] = \begin{cases} 1 & \text{if } t = t_0 \\ 0 & \text{otherwise} \end{cases}$$

Objective, cost, discomfort functions

The objective of the optimization is still to minimize the combined cost and discomfort for the prosumer; however, given the different nature of some of the input variables of this algorithm, the key functions defined in Section 3.5.1 have to be defined in a different way. In particular, the goal function is now defined as

$$\overline{G} = \overline{C} + \alpha \cdot \overline{D} \tag{3.11}$$

where $\overline{C}$ and $\overline{D}$ are the new cost and discomfort functions, and $\alpha \geq 0$ is the parameter which weights relative importance of cost and discomfort. In particular, these three functions ($\overline{G}$, $\overline{C}$ and $\overline{D}$) are defined as PDFs of random variables, so our purpose is now to minimize the expected value of $\overline{G}$.

Let us define the new cost and discomfort functions. For the first, let $\overline{\mathbf{L}}$ be an energy consumption vector, where each of its components $\overline{\mathbf{L}}[t]$ is a PDF. For each PTU t, we define the function

$$\begin{aligned} \overline{C}(\overline{\mathbf{L}})[t](x) = &\int_{-\infty}^{0} \overline{\mathbf{L}}[t](z) \cdot \overline{\mathbf{S}}[t]\left(\frac{x}{z}\right)\frac{1}{|z|}dz + \\ &+ \int_{0}^{+\infty} \overline{\mathbf{L}}[t](z) \cdot \overline{\mathbf{T}}[t]\left(\frac{x}{z}\right)\frac{1}{|z|}dz \end{aligned} \tag{3.12}$$

It has to be noted that this is a generalization of the function defined in 3.2: if we know the consumption $\mathbf{L}[t]$ beforehand at time t, the distribution $\overline{\mathbf{L}}[t]$ is the Dirac delta $\delta_{\mathbf{L}[t]}$, and consequently $\overline{\mathbf{L}}[t]$ is the Dirac delta distribution $\delta_{\mathbf{L}[t]\cdot\overline{\mathbf{T}}[t]}$. Also, in the case of ToU tariffs, $\overline{\mathbf{S}}$ and $\overline{\mathbf{T}}$ are represented by $\delta_{\mathbf{S}}$ and $\delta_{\mathbf{T}}$ respectively, in Eq. 3.12.

The new cost function $\overline{C}$ is then defined as

$$\overline{C}(\overline{\mathbf{L}}) = \sum_{t=1}^{Z} \overline{C}(\overline{\mathbf{L}})[t] \tag{3.13}$$

and in particular, it is a function that gives a PDF as an output, since each of the components defined in 3.12 is a PDF.

In order to define the new discomfort function, similar changes have to be done. About the discomfort generated by the moving of shiftable loads $\overline{D}_{shift}$, consider a shiftable load $\mathbf{v_i}$. If the load starts at the time t_0, the generated discomfort will be:

$$\overline{D}_{shift}(v_i) = \rho \cdot \sum_{t=1}^{Z} \mathbf{p_i}[t] \cdot |t_0 - t|^k \tag{3.14}$$

where, as in Eq. 3.3, ρ and $k \geq 1$ are two real numbers. Thus, with the same operations of Eq. 3.4, the total discomfort generated by the shiftable loads becomes:

$$\overline{D}_{shift} = \sum_{i=1}^{m} \overline{D}_{shift}(\mathbf{v_i}). \tag{3.15}$$

Regarding the discomfort generated by the interruptible loads, in our case it is a thermal discomfort, and, as mentioned in Section 3.5.1, it depends on the difference between the desired and the provided temperature. For this reason, given an interruptible load $\mathbf{w_i}$, in order to determine the generated discomfort $\overline{D}_{int}(\mathbf{w_i})$, the same formula seen in Eq. 3.5 is adopted, keeping in mind that each $\mathbf{w_j}$ is now a PDF, and so is $\overline{D}_{int}(\mathbf{w_j})$. With this definition, following the operations of Section 3.5.1, it follows that:

$$\overline{D}_{int} = \sum_{j=1}^{b} \overline{D}_{int}(\mathbf{w_j})$$

and finally

$$\overline{D} = \overline{D}_{shift} + \overline{D}_{int}.$$

Pseudocode of the Algorithm

In the following it is explained how the uncertainty-aware algorithm works.

1. For each shiftable load $\mathbf{v_j}$, a permitted allocation $k \in I_j$ is chosen.

2. Each $\mathbf{v_j}$ thus allocated is added to the net energy consumption vector. In other words, the following vector is created:

   $$\overline{\mathbf{L}}_{k_1,\ldots,k_m} = \overline{\mathbf{N}} + P_{k_1}(\mathbf{v_1}) + \ldots + P_{k_m}(\mathbf{v_m}).$$

   In this case, differently from the algorithm in Section 3.5.1, each of its components is a PDF.

3. The goal function relative to the total load with the chosen allocations is calculated. In this case, the vector of PDFs $\overline{G}(\overline{\mathbf{L}}_{k_1,\ldots,k_m})$ is calculated by combining Eq. 3.11, 3.13 and 3.15.

4. These three steps are repeated for every possible combination of each $k_j \in I_j$. The number of times this process is repeated is $\prod_{j=1}^{m} |I_j|$. Similarly to the algorithm discussed in Section 3.5.1, it is possible to run the steps described so far in parallel for each possible choice of the k_j.

5. As in the algorithm in Section 3.5.1, the purpose is to choose the allocations that achieve the lowest value for the goal function. In this case the goal function has not a number as output, but a PDF: we chose the expected value. We then consider the $k_1, \ldots, k_m$ which minimize the expected value of the goal function $\overline{G}(\overline{\mathbf{L}}_{k_1,\ldots,k_m})$.

6. A check is performed in order to ensure that energy consumption with the chosen allocation does not exceed the maximum load limit at any PTU. However, each component of $\overline{\mathbf{L}}_{k_1,\ldots,k_m}$ is a PDF and M is a number, so these inequalities have to be defined in another way. The load limit might exceed with a certain probability. We chose a number $p \in (0,1)$, and defined the bounds as

$$\int_{-\infty}^{M} \overline{\mathbf{L}}_{k_1,\ldots,k_m}[t] \leq p \tag{3.16}$$

for each PTU t. In other words, at each PTU, the probability of the load not exceeding the limit is at least p. If $\overline{\mathbf{L}}_{k_1,\ldots,k_m}(t)$ satisfies this condition for every $t \in \{1, \ldots, Z\}$, these values for k_j are defined as optimal; otherwise this configuration is discarded and the algorithm goes back to the previous step, taking the next best choice of the k_j.

7. Interruptible loads are modified depending on the PTUs where tariff changes, similarly to the uncertainty-unaware algorithm. The only difference is that, with the same notation, Q_j^1 is requested to satisfy the constraints of not exceeding the energy output of the device and the load limit with a relatively high probability, in a way similar to Eq. 3.16.

8. The output of the algorithm is represented by the optimal values for the k_j, and the values of Q_j.

This algorithm takes into consideration many more factors than the one in Section 3.5.1; for this reason, the computational complexity is higher. For this algorithm and the one in Section 3.5.1 the number of times the goal function is calculated is $\prod_{j=1}^{m} |I_j|$, and a check is performed afterwards to see that all the constraints are verified. However, all these operations are computationally more expensive for the uncertainty-aware algorithm, since in this case operations are made on probability distributions instead of vectors, and integrals have to be computed. More details about the parameters and settings of this algorithm will be given in Section 3.6.

Example

To explain the concept, imagine the data and flexibility settings are the same of the example in Section 5.1, which are indicated in Eq. 3.8, except for $\overline{\mathbf{N}}$. The parameters are the same of Eq. 3.9.

Let the preference probability vectors be, respectively for each smart device,

$$\begin{array}{rcl}\mathbf{p_1} & = & [0.4 \quad 0.4 \quad 0.2 \quad 0 \quad 0 \quad 0 \quad 0] \\ \mathbf{p_2} & = & [0.2 \quad 0.3 \quad 0.5 \quad 0 \quad 0 \quad 0 \quad 0]\end{array}$$

Let the consumption vector be $\overline{\mathbf{N_1}}$, and defined as:

$$\overline{\mathbf{N_1}} = \begin{bmatrix} \text{Gamma}(100, 0.01) \\ \text{Gamma}(70, 0.01) \\ \text{Gamma}(70, 0.01) \\ \text{Gamma}(60, 0.01) \\ \text{Gamma}(80, 0.01) \\ \text{Gamma}(80, 0.01) \\ \text{Gamma}(110, 0.01) \end{bmatrix}$$

Moreover, let the energy production vector be $\overline{\mathbf{N_2}}$, and defined as:

$$\overline{\mathbf{N_2}} = \frac{25}{24} \cdot \text{Beta}(24, 1) \cdot \begin{bmatrix} 0.0 \\ 0.4 \\ 0.8 \\ 1.3 \\ 1.1 \\ 0.1 \\ 0.0 \end{bmatrix}$$

Therefore, $\overline{\mathbf{N}} = \overline{\mathbf{N_1}} - \overline{\mathbf{N_2}}$. Since the expected value of the sum of two variables is the sum of the expected values, by calculating the expected values of the components of $\overline{\mathbf{N_1}}$ and $\overline{\mathbf{N_2}}$, it follows:

$$\mathbf{N} = \quad [1.0 \quad 0.3 \quad -0.1 \quad -0.7 \quad -0.3 \quad 0.7 \quad 1.1]$$

where each value represents the expected value of each component of $\overline{\mathbf{N}}$.

In a similar fashion to the example in Section 5.1, the shifted loads are calculated. In particular, the first $\overline{\mathbf{L_{k_1,k_2}}}$ vectors have the corresponding expected values $\mathbf{L_{k_1,k_2}}$ as in Eq. 3.10. For each of them, each component needs to check whether it exceeds the limit M, by using Eq. (5.13). In our example, each of the components of $\overline{\mathbf{L_{k_1,k_2}}}$ satisfies the limit, except for the sixth component of $\overline{\mathbf{L_{1,1}}}$ and the seventh component of $\overline{\mathbf{L_{2,0}}}$, $\overline{\mathbf{L_{2,1}}}$, $\overline{\mathbf{L_{0,2}}}$, $\overline{\mathbf{L_{1,2}}}$ and $\overline{\mathbf{L_{2,2}}}$. Therefore these configurations are discarded, and $\overline{\mathbf{L_{0,0}}}$, $\overline{\mathbf{L_{0,1}}}$ and $\overline{\mathbf{L_{1,0}}}$ are those remaining. For each of these vectors, the goal function has to be calculated.

First, the cost is calculated. By using Eq. (5.9), it results that:

- $\overline{C}(\overline{\mathbf{L}}_{\mathbf{0},\mathbf{0}})$ has an expected value close to 5.14£;
- $\overline{C}(\overline{\mathbf{L}}_{\mathbf{0},\mathbf{1}})$ has an expected value close to 5.25£;
- $\overline{C}(\overline{\mathbf{L}}_{\mathbf{1},\mathbf{0}})$ has an expected value close to 5.16£.

where the values reported above are the two-digits decimals closest to the actual expected values.

For what the discomfort is concerned, interruptible loads behave as in the example of Section 5.1, and, thus, the expected value of the discomfort generated is approximately 0.03£ for each of the three cases. Discomfort regarding shiftable loads, however, has to be calculated again, using Eq. (5.11). It turns out:

$$
\begin{aligned}
&\overline{D}_{shift}(\overline{\mathbf{L}}_{\mathbf{0},\mathbf{0}}) = \\
&= 0.25 \cdot \left(\sum_{t=0}^{7} \mathbf{p_0}[t] \cdot |t-0|^{1.5} + \sum_{t=0}^{7} \mathbf{p_1}[t] \cdot |t-0|^{1.5} \right) \\
&= 0.25 \cdot (0.4 \cdot 0 + 0.4 \cdot 1 + 0.2 \cdot 2.82) + \\
&+ 0.25 \cdot (0.2 \cdot 0 + 0.3 \cdot 1 + 0.5 \cdot 2.82) \\
&= 0.25 \cdot (0.97 + 1.71) \approx 0.67£
\end{aligned}
$$

$$
\begin{aligned}
&\overline{D}_{shift}(\overline{\mathbf{L}}_{\mathbf{0},\mathbf{1}}) = \\
&= 0.25 \cdot \left(\sum_{t=0}^{7} \mathbf{p_0}[t] \cdot |t-0|^{1.5} + \sum_{t=0}^{7} \mathbf{p_1}[t] \cdot |t-1|^{1.5} \right) \\
&= 0.25 \cdot (0.4 \cdot 0 + 0.4 \cdot 1 + 0.2 \cdot 2.82) + \\
&+ 0.25 \cdot (0.2 \cdot 1 + 0.3 \cdot 0 + 0.5 \cdot 1) \\
&= 0.25 \cdot (0.97 + 0.70) \approx 0.42£
\end{aligned}
$$

$$
\begin{aligned}
&\overline{D}_{shift}(\overline{\mathbf{L}}_{\mathbf{1},\mathbf{0}}) = \\
&= 0.25 \cdot \left(\sum_{t=0}^{7} \mathbf{p_0}[t] \cdot |t-1|^{1.5} + \sum_{t=0}^{7} \mathbf{p_1}[t] \cdot |t-0|^{1.5} \right) \\
&= 0.25 \cdot (0.4 \cdot 1 + 0.4 \cdot 0 + 0.2 \cdot 1) + \\
&+ 0.25 \cdot (0.2 \cdot 0 + 0.3 \cdot 1 + 0.5 \cdot 2.82) \\
&= 0.25 \cdot (0.60 + 1.71) \approx 0.58£
\end{aligned}
$$

We can now conclude the example. The goal function has now an expected value of:

- $5.14£ + 0.25 \cdot (0.03£ + 0.67£) = 5.32£$ for $\overline{\mathbf{L}}_{\mathbf{0},\mathbf{0}}$
- $5.25£ + 0.25 \cdot (0.03£ + 0.42£) = 5.36£$ for $\overline{\mathbf{L}}_{\mathbf{0},\mathbf{1}}$
- $5.16£ + 0.25 \cdot (0.03£ + 0.58£) = 5.31£$ for $\overline{\mathbf{L}}_{\mathbf{1},\mathbf{0}}$

Therefore, the optimal configuration turns out to be $k_1 = 1$ and $k_2 = 0$.

3.5.3 Level 2 Optimization

As mentioned earlier in the chapter, in this level prosumers can buy/sell energy from/to other prosumers. As already shown in literature [50], and, compliant to our Cardiff data, the following assumption holds: the tariff at which prosumers without PV buy energy from the grid (Time of Use import tariff) is always higher than the tariff at which prosumers with PV sell energy to the grid (export tariff). When this happens, buying/selling energy at an intermediate price between these two tariffs is favorable for both the buyer and the seller. The optimization at this level can be generalized for RTP tariffs if the following requirements are satisfied:

- The tariffs at which prosumers buy energy from the grid are always higher than the tariffs at which they sell energy to the grid.
- There is a reliable way to predict the RTP tariffs.

Level 2 algorithm relies on the results found by level 1 algorithm. We presented two versions of the level 1 algorithm though, one which does not consider uncertainty in the data, and the other which does. For this reason, an uncertainty-aware version of the level 2 algorithm has been created as well; we decided to describe it in the same section though, since these two versions differ only for a small part. In the description of the algorithm we make the assumption of using the uncertainty-unaware version of level 1, i.e. the algorithm in Section 3.5.1. We will also explain the process when using the uncertainty-aware version of level 1 algorithm. Below it is reported the description of the related optimization algorithm.

Input of the Algorithm

First, level 1 optimization is performed on each prosumer, obtaining the optimal allocation of each shiftable load. Then, the energy consumption vector of each prosumer can be calculated by adding each obtained shiftable load to the net consumption vector. Let us call $\mathbf{N_r}$ the net consumption vector of the $r-$th prosumer and $k_1^r, \ldots, k_m^r$ the optimal allocations of the $r-$th prosumer's shiftable loads $\mathbf{v_1}^r, \ldots, \mathbf{v_m}^r$. The consumption vector of that prosumer will then be:

$$\mathbf{V_r} = \mathbf{N_r} + P_{k_1^r}(\mathbf{v_1}^r) + \ldots + P_{k_m^r}(\mathbf{v_m}^r)$$

These vectors will be the input of the level 2 algorithm, along with the tariff vectors **T** and **S** of each prosumer defined in the Section 3.5.1. In the case of RTP tariffs, **T** and **S** will be the vectors of predicted tariffs for the day, for each prosumer.

Let N be the number of users in the grid: as already mentioned in Section 3.4, in our case N is 184. From the consumption vectors, a $N \times (Z+1)$ matrix R is built such that each row of R contains a grid user's name in the first column, and the user's energy consumption at PTU t in the $t+1$-th column. Basically, each row contains the name of a grid user followed by its related energy consumption vector.

Additionally, a vector named A of size N is created and that will be used to calculate the costs at the end of the process. At the beginning, each of its components $A[r]$ is zero.

Pseudocode of the Algorithm

Here we will show how the optimization work is performed.

1. First, we consider an index t which indicates the PTU the algorithm is currently elaborating: when the algorithm begins, $t = 1$. A certain ordering of the rows of the matrix R is established, from the prosumer with the lowest selling tariff at time t, to the prosumer with the highest selling tariff at time t. The algorithm ends when $t = Z + 1$.

2. Following the order established in step 1, the values of R related to the time t (which indicate the net consumption of the prosumers at time t) are checked. When a negative number is found, then the current operation ends and the algorithm proceeds to the next step. Otherwise, t becomes $t + 1$ and the algorithm goes back to step 1.

3. Let the found negative number be c_1 and the prosumer related to that load be $Prosumer_1$. Now, the elements of R in that column are checked from the one corresponding to the grid user with the highest buying tariff at PTU t to the one with the lowest, and the search stops when either all the values have been checked, or a positive value c_2 is found. In the first case, the algorithm returns to step 1 and t becomes $t + 1$; in the second case, where the grid user relative to this value will be referred to as $Prosumer_2$, the algorithm proceeds to step 4.

4. At PTU t, $Prosumer_1$'s net consumption is negative and $Prosumer_2$'s is positive. $Prosumer_1$ then produces excess energy, and sells it to $Prosumer_2$: the amount of energy sold is either the surplus energy of $Prosumer_1$, or all the energy that $Prosumer_2$ needs at that time, depending on which is lower. In mathematical terms, this means changing the values in R so that the smallest in absolute value between c_1 and c_2 becomes zero, and the other becomes $c_1 + c_2$. Let c be the smallest between $|c_1|$ and $|c_2|$: c is the amount of transferred energy. This operation will be referred to as an *energy transaction*.

5. The price of the above transaction is calculated, using the formula

$$C = c \cdot T \tag{3.17}$$

   where C indicates the cost of the transaction and T the used tariff. Calling respectively T_1 and T_2 as $Prosumer_1$'s selling tariff and $Prosumer_2$'s buying tariff, T is defined as:

$$T = \frac{T_1 + T_2}{2} \tag{3.18}$$

   This way, both the buyer and the seller have the same profit from this process.

6. After this cost is calculated, $A[Prosumer_1]$ changes into $A[Prosumer_1] + C$, and $A[Prosumer_2]$ becomes $A[Prosumer_2] - C$. Next:
   - If the new value of c_1 is still negative, the procedure goes back to step 3.
   - If the new value of c_1 is zero, the procedure goes back to step 2.

After all the time units have been processed, this procedure stops. The algorithm's output are the matrix R and the new energy costs of all the grid users. For the $r-$th grid user in R, let us call R_r the $r-$th row of R without the first column: her energy cost after this process will be

$$C = \mathbf{R_r^+} \cdot \mathbf{T_r} + \mathbf{R_r^-} \cdot \mathbf{S_r} - A[r]$$

where the $^+$ and $^-$ signs are similar to those in the Eq. 3.2, and $\mathbf{T_r}$ and $\mathbf{S_r}$ are the purchasing and selling tariff for the $r-$th user. The components of A are the costs of internal energy transactions.

The reason why the values of R were searched with the sorting we defined in step 1 and step 3 respectively is to guarantee to the prosumers the highest possible profit from the performed transactions. In particular, in Eq. 3.18, it can be seen that the profit for the seller depends on how high the tariff $\mathbf{T_2}$ for the buyer is, and the amount of money saved for the buyer depends on how low the tariff $\mathbf{T_1}$ for the seller is. For these reasons, it is more profitable for the seller to sell energy to the prosumers with the highest purchasing tariffs, and for the buyer it is more profitable to buy energy from the sellers with the lowest selling tariffs. This proves that the total amount of savings that prosumers obtain from level 2 is maximized by operating in the order described by the algorithm.

When this phase ends, a check is performed on the consumption: for each PTU, the sum of the power consumption relative to that time is checked in order to see if the maximum capacity of the grid is exceeded. In our tests the maximum of these values barely exceeded $200kW$, and only for the most critical settings it may go slightly over $300kW$, while grid capacity is always higher than that.

Dealing with Uncertainty

In the case when uncertainty is considered, the steps of the algorithm remain very similar but with a minor change: every component of the consumption vectors is represented as a PDF, and so are the elements of R. Also, as additional input, the PV generation of all the prosumers will be needed. Below we report a brief description of such an algorithm.

- The first and second steps proceed in a similar way: in the second step, at the current PTU, the expected value of each grid user's consumption is checked. In other words, the numbers checked by the algorithm are the expected values of each element of R (remind that, in this case, elements of R are PDFs): they are checked in the same order as the optimal version algorithm. The algorithm proceeds to the next step when a negative value is found.

- In the third step, let $p \in (0,1)$ be a real number, c_o be the negative value found in step 2, $Prosumer_1$ the prosumer relative to that value, and c_m the highest number so that the probability of $Prosumer_1$ generating c_m kWh energy or more at that PTU is at least p. To give a better definition of c_m, let Pv be the PDF representing PV generation for $Prosumer_1$: c_m is then defined as the maximum number such that

$$\int_{c_m}^{+\infty} Pv(x)dx \geq p.$$

  The third step of the optimization proceeds like the optimal version algorithm, with c_1 being the highest value between c_o and c_m. This limitation has been included in order to ensure that $Prosumer_1$ has probability at least p to be able to generate the energy that will be sold to the other grid users.

Every other step is identical to the optimization without uncertainty. The computational complexity is calculated as follows. The number of checks for negative entries in step 1 is $N \cdot Z$, and for each negative value, at most $N-1$ values of the same column are checked in step 2. For each of these cases, the operations performed in the next steps are constant (one minimum evaluation between two values, four sums, one product and one arithmetic mean between two values). In the uncertainty-aware algorithm, for each of the cases detected in step 2, there is an additional check which requires the calculation of an integral.

Example

Imagine that, after level 1 optimization, there are three prosumers with the following consumption vectors:

$\mathbf{v_1}$	0.2	0.5	0.4	−0.1	0.3	0.7	1.0
$\mathbf{v_2}$	−0.3	0.0	0.1	1.1	0.2	1.0	0.0
$\mathbf{v_3}$	0.2	−0.4	0.3	0.3	0.0	0.5	1.1

where each energy value is expressed in kWh. Let us assume that the selling tariff is 0.3 £/kWh at each PTU for each prosumer: since the tariff is the same for every prosumer, we can perform step 2 without having to establish a sorting in step 1. The buying tariff vectors are the following, expressed in £/kWh:

$\mathbf{T_2}$	1.0	1.0	1.0	1.0	1.2	1.2	1.0
$\mathbf{T_3}$	0.8	0.8	0.8	0.8	1.1	1.1	0.8
$\mathbf{T_1}$	0.7	0.7	0.7	0.7	0.9	0.9	0.7

It can be noticed that the order of the selling tariffs, from the highest to the lowest, is the same at each PTU. For this reason, at the third step, the first values to be analyzed will be the ones relative to $\mathbf{v_2}$, then to $\mathbf{v_3}$ and finally to $\mathbf{v_1}$.

When the algorithm starts, it first checks PTU 1 and detects that it has an energy surplus for prosumer 2 since the net consumption is negative. For that PTU, a prosumer that can buy that energy is searched.

First $\mathbf{v_3}$ is analyzed, and then $\mathbf{v_1}$, since $\mathbf{v_3}$'s tariff is higher at PTU 1. It turns out that $\mathbf{v_3}$ consumes 0.2 kWh at PTU 1 and, since $\mathbf{v_2}$ is selling 0.3 kWh, $\mathbf{v_3}$ may buy all of the energy she needs from $\mathbf{v_2}$.

The transaction takes place, using the tariff in Eq. (5.15): $\mathbf{v_2}$ sells 0.2 kWh with the tariff $\frac{0.3+0.8}{2} = 0.55$ £/kWh to $\mathbf{v_3}$. The transaction cost is then the product of the energy sold and the tariff as in Eq, (5.14): in this case, $0.2 \cdot 0.55 = 0.11$ £.

Also, since the transaction tariff is the arithmetic mean of the purchasing and selling tariffs, the gain will be the same for both the prosumers. Using Eq. (5.14) it can be noticed that if the buyer bought 0.2 kWh energy from the grid, it would have cost $0.2 \cdot 0.8 = 0.16$ £, since her tariff at PTU 1 is 0.8: so, buying energy from the other prosumer produces a saving of $0.16 - 0.11 = 0.05$ £. Also, using (5.14), it results that the seller would have sold energy at price $0.2 \cdot 0.3 = 0.06$ £, and the related revenue will increase by $0.11 - 0.06 = 0.05$ £.

After this process, R has changed. The new R is:

$\mathbf{v_1}$	0.2	0.5	0.4	−0.1	0.3	0.7	1.0
$\mathbf{v_2}$	**−0.1**	0.0	0.1	1.1	0.2	1.0	0.0
$\mathbf{v_3}$	**0.0**	−0.4	0.3	0.3	0.0	0.5	1.1

Since the value of $\mathbf{v_2}$ at PTU 1 is still negative, the algorithm resumes from step 3. Now $\mathbf{v_2}$ has still 0.1 kWh to sell, and $\mathbf{v_1}$ can buy up to 0.2 kWh. Therefore, $\mathbf{v_2}$ will sell 0.1 kWh to $\mathbf{v_1}$. After doing so, $\mathbf{v_2}$ will have no more energy to sell, and the algorithm will go to step 2, and then to step 1 since there are no more energy sellers at PTU 1.

The algorithm then proceeds similarly for each PTU. The other transactions will occur at PTU 2, where $\mathbf{v_3}$ will sell the excess energy, and at PTU 4, where the seller will be $\mathbf{v_1}$. After all the calculations, the final configuration will be:

$\mathbf{v_1}$	**0.1**	**0.1**	0.4	**0.0**	0.3	0.7	1.0
$\mathbf{v_2}$	**0.0**	0.0	0.1	**1.0**	0.2	1.0	0.0
$\mathbf{v_3}$	**0.0**	**0.0**	0.3	0.3	0.0	0.5	1.1

It turns out that $\mathbf{v_1}$'s total energy cost is 2.245 £, with a saving of 0.095 £; $\mathbf{v_2}$'s total cost is 2.445 £, with a saving of 0.105 £ and $\mathbf{v_3}$'s total cost is 1.86 £, with a saving of 0.09 £.

3.5.4 Level 2 Optimization - Parallel version

In the previous subsection, level 2 optimization has been formulated as a sequential process: buyers and sellers are chosen one at a time, and their cooperation will influence future interactions that will happen between other buyers and sellers. However, the exploitation of parallel calculation can greatly reduce the computational time. As such, we have invested efforts on this aspect by (i) identifying some parts of the previously described level 2 optimization which can be executed in parallel, and (ii) formulating, accordingly, a slightly different version of the level 2 algorithm.

Parallelizable sections of the serial algorithm

The following sections of the level 2 optimization described in Section 3.5.3 present some aspects where parallel calculation can be employed.

- Preliminary level 1 optimization is executed separately for each grid user, and optimizations for different users do not influence each other. Consequently, this process can be run simultaneously for each user.
- Transactions which happen at a certain time unit do not influence transactions happening at other time units. For this reason, the algorithm described in Level 2 optimization can be executed separately for each time unit.
- Once each transaction has been defined in terms of which users are involved and how much energy is being transferred, the calculation regarding payments and earnings can be operated simultaneously for each couple of users.

With the employment of the CUDA programming it is possible to efficiently perform these tasks, with significant reductions in computational time compared to the algorithm described in Section 3.5.3.

Parallel version of the algorithm

In order to have a more favorable context for employing parallel computation in level 2 optimization, we designed a different version of the algorithm whose behavior is better suited for this purpose. The input of this algorithm is the same of the one described in Section 3.5.3 with the exception of A, the vector used for calculating the costs. For this purpose we will use a $N \times Z$ matrix instead, which we will call $A^{'}$ for this subsection.

This is the pseudocode of the parallel version of level 2 algorithm.

1. First, we consider the index t which represents the PTU we are working on. Since this algorithm can be run for each PTU independently, this and the following steps are processed in parallel for each possible choice of t.
2. Let t_c be the total consumption of energy through the grid at time t: in other words, t_c is the sum of all the positive components of R in its $t-$th column. Let t_p be the total production of energy through the grid at time t, that is, the sum of all negative components of R in its $t-$th column changed by sign. Our goal is to create a vector E of size N, such that the $r-$th component $E[r]$ represents the amount of energy that the $r-$th grid user will buy (if positive) or sell (if negative). Consequently, we define E as it follows:

- If $t_p > t_c$, users are considered from the one with the lowest selling tariff to the one with the highest selling tariff. Let S_p be the sum of all the negative components of E: S_p will change as E is updated. Considering users in the order described above, and identifying by r the current user:
  - if r is such that $R[r, j]$ is negative and $S_p + R[r, j] < -t_c$, then $E[r] = -t_c - S_p$;
  - if r is such that $R[r, j]$ is negative and $S_p + R[r, j] \geq -t_c$, then $E[r] = R[r, j]$.

  Afterwards, the users are considered from the one with the highest buying tariff to the one with the lowest buying tariff. Following this order, if r is such that $R[r, j]$ is positive, we define $E[r] = R[r, j]$.
- If $t_p < t_c$, users are considered from the one with the highest buying tariff to the one with the lowest buying tariff. Let S_c be the sum of all the positive components of E: S_c will change as E is updated. Considering users in the order described above, and identifying by r the current user:
  - if r is such that $R[r, j]$ is positive and $S_c + R[r, j] > t_p$, then $E[r] = t_p - S_c$;
  - if r is such that $R[r, j]$ is positive and $S_c + R[r, j] \leq t_p$, then $E[r] = R[r, j]$.

  Afterwards, the users are considered from the one with the lowest selling tariff to the one with the highest selling tariff. Following this order, if r is such that $R[r, j]$ is negative, we define $E[r] = R[r, j]$.

3. For the sake of notation, let $T[r]$ and $t[r]$ be, respectively, the buying and the selling tariffs associated to the user with index r. Let us define the following quantities:

$$S_0 = \sum_{j:E[j]>0} E[j] \cdot T[j]$$
$$S_1 = -\sum_{j:E[j]<0} E[j] \cdot t[j].$$

4. This step is run in parallel for each user. Let r be the index associated to the user we are considering.

   - If $E[r] > 0$, the cost for the user is determined by the formula

$$A^{'}[r, j] = E[r] \cdot T[r] \cdot \frac{S_0 + S_1}{2 \cdot S_0}. \tag{3.19}$$

   - If $E[r] < 0$, the profit for the user is determined by the formula

$$A^{'}[r, j] = E[r] \cdot t[r] \cdot \frac{S_0 + S_1}{2 \cdot S_1} \tag{3.20}$$

5. This step is run in parallel for each user. The matrix R has its values updated. Since $E[r]$ represents the amount of energy bought/sold by the user corresponding to the index r, R will become:

$$R[r, j] = R[r, j] - E[r].$$

The output of this algorithm are the two matrices R and A'. For each user, calling r the corresponding index, their energy cost after this process will be

$$C = \mathbf{R}_\mathbf{r}^+ \cdot \mathbf{T}_\mathbf{r} + \mathbf{R}_\mathbf{r}^- \cdot \mathbf{S}_\mathbf{r} + \sum_{j=1}^{Z} A'[r, j]$$

where the $^+$ and $^-$ signs and the vectors $\mathbf{T}_\mathbf{r}$ and $\mathbf{S}_\mathbf{r}$ are described in Section 3.5.3.

There are two reasons for this choice of the functions described in Eq. 3.19 and Eq. 3.20. The first is that the total amount of saved money among the buyers and the total amount of money gained among the sellers is the same as in the algorithm described in Section 3.5.3. The second reason is that these tariffs allow the buyers to purchase energy at a lower tariff than that used to sell energy to the grid, and guarantee the sellers to trade energy at a higher tariff than that used to buy energy from the grid.

3.5.5 Level 3 Optimization - External Input Algorithm

In the following we will describe the algorithm for the level 3, with the assumption that level 2 is performed first so that shiftable loads are already in optimal positions for the prosumers, and the energy produced internally to the grid is already exploited. Also for this reason, this algorithm can have two versions, with and without uncertainty in data. Both versions can greatly benefit from the parallel computation employed for the preliminary level 1 and 2 optimizations. We start by describing the algorithm where uncertainty is not considered.

Input of the Algorithm

The input for this optimization consists of:

- **Consumption of all the users.** More precisely, the matrix R described in Section 3.5.3.
- **Shiftable and interruptible loads.** Since level 1 optimization has already been performed, it is important to know how the shiftable load vectors are allocated for each prosumer, and how and when interruptible loads are active.
- **Tariffs, maximum load, flexibility settings and user preferences.** These entities have already been described in Section 3.5.1. In the case of RTP tariffs this algorithm is not necessary, since the needs of the External Party (*EP*) are already fulfilled by the level 1 optimization.

Moreover, some other parameters have to be inserted manually by the EP, as shown later in the section.

After the first two levels optimization has been performed, the EP may check the grid's total load at each PTU, and must choose the PTU t to shift loads from. Finally, the EP must choose how much energy to move and which kind of load shifting to perform. There are two choices.

- In the first one, each shiftable load currently allocated with a consumption at t different than zero, is reallocated in a position where its consumption at t is zero. Among all the positions that satisfy this requirement, the one which minimizes the prosumer's daily energy cost is chosen. Moreover, each interruptible load is decreased at time t and its energy output is increased in adjacent PTUs, provided it does not fail the power limit constraints.
- In the second one, each shiftable load currently allocated with a consumption at t different from zero is moved by a number of PTUs chosen by the EP and according to the prosumers' flexibility settings.

Pseudocode of the Algorithm

If the first type of load shifting is chosen:

1. Each shiftable load whose component at t is different than zero is shifted in every possible position according to the flexibility settings of the related prosumer, as described in Section 3.5.2. If either the shiftable load in the new position has the component at t greater than zero, or the prosumer's load exceeds the load limit at any PTU after shifting the load in the new position, that position is discarded.
2. For each available position, the goal function obtained combining cost and discomfort as described in Eq. 3.1 is calculated: the new candidate position for the shiftable load will be the one where the goal function reaches its minimum.
3. For each processed shiftable load, the algorithm calculates how much the value of the goal function increases for the corresponding prosumer if it is moved in its candidate position: this increase will be called *operation cost*. Then, the shiftable loads are moved, from the one with the lowest operation cost to the one with the highest, until either all the load shifts have been made or enough energy has been moved from the time t to fulfill EP's request. Then, the sum of these shifts' operation costs is calculated.
4. For each interruptible load whose component at t is different from zero, its output at $t+1$ and $t-1$ is increased by the maximum quantity, under the following constraints:
   - the load limit is not exceeded at $t+1$ and $t-1$;
   - let P be the combined increase of power at times $t+1$ and $t-1$: P must not be higher than the power output of the interruptible load at time t.

   After doing so, the power output of the interruptible load at time t is reduced by P. If the power output at t is zero, this procedure stops. Otherwise, the EP repeats the procedure by moving the load at times $t+2$ and $t-2$ until it is no more convenient or possible to move loads (e.g. the power output at t is zero).

If the second type of load shifting is chosen:

1. Each shiftable load whose component at t is different from zero is shifted by a certain number of PTUs, according to EP's request. However, if either the prosumer's flexibility settings do not allow the shift or the prosumer's maximum load is exceeded at any PTU, that load cannot be moved.
2. For the loads that can be moved, the new combinations of cost and discomfort and the related operation costs are calculated according to Eq. 3.1.
3. Again, the load shifts are performed, from the one with the lowest operation cost to the one with the highest, until either all the load shifts have been made or enough energy has been moved at time t to fulfill EP's needs. The sum of all those shifts' operation costs is calculated.

Once this process ends, the cost for the EP has to be calculated. It corresponds to the operation cost for each involved prosumer, so that the prosumer is refunded for the increase in cost or discomfort; however, different ways of refunding prosumers can be used, as seen for example in [31]. Then, the EP decides whether to proceed with the shifting or not. If yes, economic transactions occur and loads are shifted as requested. If not, no transactions occur and the procedure starts again.

Dealing with Uncertainty

Regarding the version of the algorithm which uncertainty, the procedure is mostly the same. The differences are relative to the fact that elements of R and the goal function are PDFs, as for the uncertainty case of level 2 optimization described in Section 3.5.3. Thus, the operation cost will be the increase for the expected value of the goal function. The check for not exceeding the limit is performed in the usual way: fixed a number $p \in (0,1)$, it is valid if the probability of the PDF not exceeding the limit is at least p, as in Eq. 3.16.

Example

As usual, energy will be expressed in kWh and tariffs in £/kWh. Suppose that after level 1 and level 2 optimizations there are two prosumers, with the following consumption vectors:

$$\begin{array}{cccccccc} \mathbf{v_1} & 1.2 & 1.5 & 2.4 & 2.1 & 0.3 & 0.7 & 1.0 \\ \mathbf{v_2} & 1.1 & 2.0 & 2.1 & 2.5 & 0.8 & 0.4 & 0.7 \end{array} \tag{3.21}$$

Let us assume each of the two prosumers has a shiftable load:

$$\begin{array}{cccccccc} \mathbf{f_1} & 0.0 & 0.0 & 0.3 & 0.4 & 0.0 & 0.0 & 0.0 \\ \mathbf{f_2} & 0.0 & 0.0 & 0.1 & 0.3 & 0.1 & 0.0 & 0.0 \end{array} \tag{3.22}$$

Also, let us suppose the maximum energy consumption allowed is 3 kWh at each PTU for both users, and that flexibility settings allow each shiftable load being moved up to two time units forward, or up to two time units backward. Finally, let us assume discomfort settings are the same of Eq. 3.9.

The total load vector is then $\mathbf{v} = \mathbf{v_1} + \mathbf{v_2}$, that is:

v	2.3	3.5	4.5	4.6	1.1	1.1	1.7

Let us assume that the *EP* chooses the PTU 4, where the maximum total load occurs with a load of 4.6 kWh. Therefore, it might want to use our level 3 optimization to shave loads at PTU 4. Let us suppose it chooses to move the loads by a certain number of PTUs, in this case two units forward. The new loads are, in terms of operations:

$$\mathbf{v_r}(new) = \mathbf{v_r}(old) - \mathbf{f_r} + P_k(\mathbf{f_r}) \tag{3.23}$$

where P_k (in this case, $k = 2$) is the rotation defined in Section 5.1. In this case, the rotated vectors are

$P_2(\mathbf{f_1})$	0.0	0.0	0.0	0.0	0.3	0.4	0.0
$P_2(\mathbf{f_2})$	0.0	0.0	0.0	0.0	0.1	0.3	0.1

And the new load vectors become:

$\mathbf{v_1}$	1.2	1.5	2.1	1.7	0.6	1.1	1.0
$\mathbf{v_2}$	1.1	2.0	2.0	2.2	0.8	0.7	0.8
v	2.3	3.5	4.1	3.9	1.4	1.8	1.8

and, as expected, this allowed us to reduce the maximum load at PTU 4. Also, since requirements on flexibility settings and maximum loads are satisfied, these shiftings are allowed.

Let us now consider the cost of this operation for the EP. Let us suppose both prosumers have the same buying tariff, represented by the vector:

T	1.0	1.0	0.5	0.5	0.5	1.2	1.2

(3.24)

The selling tariff is not needed, since neither of the vectors $\mathbf{v_1}$ and $\mathbf{v_2}$ has negative components. It can be easily calculated by Eq. (5.2) that energy cost was 7.14 £ for the first prosumer and 7.12 £ for the second prosumer, but after our optimization it can be calculated, using the same formula, that they became 7.42 £ and 7.40 £ respectively. Thus, since the prosumers would pay more ($7.42 - 7.14 = 0.28$ £ the first, $7.40 - 7.12 = 0.28$ £ the second), they have to be refunded of their expenses in order to give the requested flexibility.

However, energy cost is not the only thing that has to be refunded. The operation requested by the EP may also cause an increase of their discomfort which should be covered too. In order to quantify the increase of discomfort, we calculate it before and after the requested load shifting. Using the formula in Eq. (5.3), the initial discomfort can be expressed as:

$$D_{shift}(\mathbf{v_1}) = 0.25£ \cdot |2-2|^{1.5} = 0£$$
$$D_{shift}(\mathbf{v_2}) = 0.25£ \cdot |2-1|^{1.5} = 0.25£$$

while after the requested load shifting, it becomes

$$D_{shift}(\mathbf{v_1}) = 0.25£ \cdot |4-2|^{1.5} \approx 0.71£$$
$$D_{shift}(\mathbf{v_2}) = 0.25£ \cdot |4-1|^{1.5} \approx 1.30£.$$

These values have to be rescaled to their relative importance in the optimization. To be more precise, since the importance of discomfort compared to the cost is decided in the goal function (defined in Eq. (5.1)) by the parameter α, these values have to be rescaled by α. In other words, the economic value of the prosumers' discomfort is $0.25 \cdot 0.71£ \approx 0.18£$ for the first prosumer, and $0.25 \cdot 1.30£ \approx 0.33£$ for the second.

Hence, it is now possible to calculate the rewards the EP would need to pay to each of them.

- For the first prosumer, the increase of cost is worth 0.28 £ and the discomfort would be quantified to 0.18 £, which means she has to be refunded by 0.46 £ by the EP.
- For the second prosumer, the increase of cost is worth 0.28£, while the increase of discomfort has been calculated to be the equivalent of 0.33£ which indicates a reward of 0.61£ for the second prosumer.

3.5.6 Level 3 Optimization - Automatic Algorithm

As the EP actively provides input to the level 3 optimization algorithm, in this section we discuss a version of it where the input is automatically chosen in order to minimize the peak load. Similarly to the algorithm described in Section 3.5.5, the computational time of this version can greatly benefit from the parallel computation when employed for the preliminary level 1 and 2 optimization.

Input of the Algorithm

This algorithm has the same input of the one seen in Section 3.5.5, except for the parameters chosen by the EP which are automatically computed. Again, in case of RTP tariffs, the whole algorithm is not used, since the purpose of peak shaving is already realized at level 1. Below, this procedure is briefly described.

Pseudocode of the Algorithm

1. First, the total energy consumption of the grid at each PTU is computed. Let t be the PTU with the highest energy consumption.

2. The shiftable loads of a random prosumer P_1 are chosen to check if at least one of their values at PTU t is different from zero. When this occurs, the algorithm proceeds with step 3. Otherwise the algorithm ends: in this case, peak shaving by using flexibility is not possible.

3. For each possible allocation of P_1's shiftable loads, the total energy consumption that the grid would have at each PTU and the cost-discomfort goal function are calculated. Also, the load limit for the prosumer is checked, as in Eq. 3.7: if the new allocation exceeds this limit, the allocation is discarded.

4. Among the tested positions not discarded, the load is reallocated in the combination which minimizes the highest total load among all PTUs. If there is more than one such a combination, the algorithm chooses that with the lowest value for the prosumer's goal function. The algorithm is repeated and eventually ends at step 2.

It should be noted that this algorithm is guaranteed to end, since at each iteration either the maximum total load or the goal function of a prosumer becomes strictly lower than their previous value.

Moreover, in this version of the algorithm, obtaining the highest load reduction has priority over the costs. We will see in Section 3.6.3 its results in economical terms and in terms of peak load reduction.

As far as the uncertainty is concerned, it can be applied similarly as seen in Section 3.5.5. In terms of computational complexity, this algorithm requires at each iteration one check for the PTU with the maximum consumption, the computation of the consumption for each possible shifting of the flexible loads that can be moved from that PTU, and for each of these loads, one more check for the shifting which guarantees the lowest peak. The number of iterations is limited by the number of flexible loads in the grid multiplied by Z, and at each iteration the number of consumption vectors that have to be calculated is less or equal than the number of flexible loads in the grid multiplied by Z.

Example

Suppose the data are the same of the example in Section 5.4: that is the load vectors are given in Eq. 3.21 and the shiftable loads in Eq. 3.22. Suppose the flexibility settings are also the same: in other words, loads may be shifted up to two time units forward or backward, and the tariff vector for both prosumers is the one in Eq. 3.24. Also, the discomfort settings are the ones described in Eq. 3.9.

As seen before, our total load vector is:

$$\mathbf{v} \quad 2.3 \quad 3.5 \quad 4.5 \quad 4.6 \quad 1.1 \quad 1.1 \quad 1.7$$

It follows from the first step that $t = 4$. As shown in Eq. 3.22 the loads to be shifted are $\mathbf{f_1}$ and $\mathbf{f_2}$.

The procedure starts with $\mathbf{f_1}$. Recall that if a shiftable load $\mathbf{f_r}$ is moved, the formula to determine the new load $\mathbf{v_r}$ is obtained by Eq. 3.23. Thus, fixing $r = 1$ it results:

$\mathbf{v_1}(-2)$	1.5	1.9	2.1	1.7	0.3	0.7	1.0
$\mathbf{v_1}(-1)$	1.2	1.8	2.5	1.7	0.3	0.7	1.0
$\mathbf{v_1}(0)$	1.2	1.5	2.4	2.1	0.3	0.7	1.0
$\mathbf{v_1}(1)$	1.2	1.5	2.1	2.0	0.7	0.7	1.0
$\mathbf{v_1}(2)$	1.2	1.5	2.1	1.7	0.6	1.1	1.0

which also changes the total load vector accordingly. It becomes

$\mathbf{v}(-2)$	2.6	3.9	4.2	4.2	1.1	1.1	1.7
$\mathbf{v}(-1)$	2.3	3.8	4.6	4.2	1.1	1.1	1.7
$\mathbf{v}(0)$	2.3	3.5	4.5	4.6	1.1	1.1	1.7
$\mathbf{v}(1)$	2.3	3.5	4.2	4.5	1.5	1.1	1.7
$\mathbf{v}(2)$	2.3	3.5	4.2	4.2	1.4	1.5	1.7

The highest total load is 4.2 kWh for $k = -2$ and $k = 2$, 4.6 kWh for $k = -1$ and $k = 0$ and 4.5 kWh for $k = 1$. Among these, the lowest value is chosen, that is for $k = -2$ and $k = 2$. It turns out that the lowest increase among $k = -2$ and $k = 2$ for the goal function takes place for $k = 2$ with a 0.37 £ increase, so $\mathbf{f_1}$ will be rotated forward by two positions.

For $\mathbf{f_2}$ the same procedure is applied, taking into account that $\mathbf{f_1}$ has already been moved. Thus $\mathbf{v}$ becomes:

$\mathbf{v}$	2.3	3.5	4.2	4.2	1.4	1.5	1.7

As mentioned before, $\mathbf{f_2}$ can be moved up to two positions forward or backward. The total load vector, obtained by Eq. 3.23, becomes the following:

$\mathbf{v}(-2)$	2.4	3.8	4.2	3.9	1.3	1.5	1.7
$\mathbf{v}(-1)$	2.3	3.6	4.4	4.0	1.3	1.5	1.7
$\mathbf{v}(0)$	2.3	3.5	4.2	4.2	1.4	1.5	1.7
$\mathbf{v}(1)$	2.3	3.5	4.1	4.0	1.6	1.6	1.7
$\mathbf{v}(2)$	2.3	3.5	4.1	3.9	1.4	1.8	1.8

The highest total load is 4.1 kWh for $k = 1$ and $k = 2$, 4.2 kWh for $k = -2$ and $k = 0$, and 4.4 kWh for $k = -1$. The lowest value among these is obtained for $k = 1$ and $k = 2$ and the lowest increase for the goal function among them is obtained with $k = 1$ and is equal to 0.11 £, so $\mathbf{f_1}$ will be shifted forward by one position. Therefore, the only thing left is to calculate the rewards for the prosumers. In this case the refund for the first prosumer will be 0.37£, while for the second prosumer will be 0.11£.

Therefore, $\mathbf{v}$ becomes equal to $\mathbf{v}(1)$ which corresponds to

$\mathbf{v}$	2.3	3.5	4.1	4.0	1.6	1.6	1.7

and the new shiftable loads are

$\mathbf{f_1}$	0.0	0.0	0.0	0.0	0.3	0.4	0.0
$\mathbf{f_2}$	0.0	0.0	0.0	0.1	0.3	0.1	0.0

The procedure for PTU 4 ends. The new maximum load becomes 4.1 kWh, at PTU 3, and the procedure therefore starts over considering the shiftable loads $\mathbf{f_1}$ and $\mathbf{f_2}$ different than zero at PTU 3. Since they are zero, the procedure ends.

3.6 Results and Discussion

In this section we will show the results of our algorithms on the data described in Section 3.4. In particular, Section 3.6.1 shows the results for the deterministic and uncertainty-aware versions of the algorithm for level 1 optimization in terms of savings and computational time with respect to the performed baseline algorithm Cardiff data come along. Section 3.6.2 shows the impact of the level 2 optimization, whereas Section 3.6.3 shows the automatic version of level 3 algorithm. Experiments have been run on a x64-based PC with 16GB of RAM and a 4-core processor with 2.81GHz; scripts have been written in Python language and their parallelized version run on a NVidia TitanX GPU[5]. The following libraries have been used: *xlrd* to extract data from Microsoft Excel[6], *NumPy* to operate with vectors[7], *SciPy*[8] and *PaCAL*[9] for random variables manipulation. Furthermore, the *numba*[10] compiler has been used for CUDA[11] programming.

3.6.1 Level 1 Optimization

A setup is a combination of one prosumer and five settings. For example, the following three combinations define three different setups:

- HOUSE A4, 2020, *Green*, *high*, *spring*, *sunday*
- HOUSE A5, 2020, *Green*, *high*, *spring*, *sunday*
- HOUSE A4, 2020, *BAU*, *high*, *winter*, *sunday*

The first element indicates the prosumer, the others indicate the five types of settings already defined within Section 3.4.

Level 1 algorithms have been tested on all the prosumers for a total of 11415 different setups where each prosumer had at least one shiftable load.

Deterministic Algorithm

The deterministic algorithm generates savings for 5680 setups (49.76% of the total number of setups). Table 3.3 shows the details of the corresponding results. In particular, the columns indicate the number of shiftable loads.

In the following we will explain what each row in the table indicates:

1. Number of setups tested. Overall number of prosumers with one (two and three) shiftable load(s) only for each different combination of setups.

[5]Please contact the author to obtain the developed scripts and the used data.
[6]`https://pypi.python.org/pypi/xlrd/`
[7]`https://pypi.org/project/numpy/`
[8]`https://pypi.org/project/scipy/`
[9]`https://pypi.org/project/PaCal/`
[10]`https://numba.pydata.org/`
[11]`https://developer.nvidia.com/cuda-zone`

2. Average value of the goal function with the naive algorithm.

3. Average value of the goal function after level 1 optimization.

4. Average reduction of the goal function when the level 1 optimization has been performed.

5. Average reduction in % of the goal function when the level 1 optimization has been performed.

6. Average computational time for the non-parallel version of the algorithm.

7. Average computational time for the parallel version of the algorithm.

		1 Load	2 Loads	3 Loads
1	Setups	6194	2924	1829
2	AvgCost Naive	0.57£	2.00£	3.59£
3	AvgCost Opt	0.45£	1.62£	2.60£
4	Avg Savings	0.12£	0.38£	0.99£
5	Avg %Savings	21.05%	19.00%	27.58%
6	Non-CUDA Time	0.14s	0.53s	8.90s
7	CUDA Time	0.15s	0.16s	0.18s

Table 3.3: Results of level 1 optimization in terms of cost-discomfort saving, and computational time. Costs are calculated on a 24 hours time interval.

Uncertainty-aware Algorithm

The uncertainty-aware algorithm generates savings for 9945 setups (87.12% of the total number of setups). This result is higher than the one obtained with the deterministic algorithm. This happens because in the deterministic case the preference of the prosumer is a specific PTU, thus if the optimization matches this preference there is no saving. On the other hand, in the uncertainty-aware algorithm user preferences are defined by a probability for each PTU, and for this reason expected value of savings is usually positive.

Detailed results on this algorithm are described in Table 3.4. In particular, each row of the table has the same description of Table 3.3.

The reader notices that only a few setups included four or more loads: that is why we have not included them in the results. Anyway, behaviors of four or more loads and three loads are comparable.

		1 Load	2 Loads	3 Loads
1	Setups	6194	2924	1829
2	AvgCost Naive	2.26£	2.66£	4.51£
3	AvgCost Opt	1.75£	1.98£	3.20£
4	Avg Savings	0.51£	0.68£	1.31£
5	Avg %Savings	22.57%	25.56%	29.05%
6	Non-CUDA Time	2.86s	4.50s	10.77s
7	CUDA Time	0.20s	0.22s	0.25s

Table 3.4: Results of *level 1 - uncertainty* optimization in terms of cost-discomfort saving, and computational time. Costs are calculated on a 24 hours time interval.

3.6.2 Level 2 Optimization

We recall that, while level 1 optimization is related to a single prosumer, level 2 optimization is related to all the users in the grid. In this case, we have run our experiments for each possible settings configuration, for a total of 180 combinations as reported in Section 3.4.

We have collected results on the number of transactions between prosumers, average savings and computational time, and reported the average on the 180 combinations on Table 3.5. In particular, each row describes the following metric:

1. *Avg Trans*: Describes the average number of energy transactions among prosumers in a 24h time interval.

2. *Avg # Prosum*: Describes the average number of prosumers involved within energy transactions.

3. *Avg Savings*: Describes the average savings for each prosumer.

4. *Avg Savings (Prod)*: Describes the average savings for prosumers who are energy producers.

5. *Avg Savings (NProd)*: Describes the average savings for prosumers who are not energy producers.

6. *CUDA Time lev 1*: Describes the average computational time when CUDA optimization is used only for each preliminary level 1 computation.

7. *CUDA Time lev 1-2*: Describes the average computational time when CUDA optimization is used for both level 1 and level 2 optimizations, as described in Section 3.5.4.

Results depend heavily on PV production in the grid because the greater the production of energy, the higher the number of transactions. Therefore, settings allowing a better

		Results
1	*Avg Trans*	**1115.63**
2	*Avg # Prosum*	**48.97**
3	*Avg Savings*	**1.31£**
4	*Avg Savings (Prod)*	**1.80£**
5	*Avg Savings (NProd)*	**0.99£**
6	*CUDA Time lev 1*	**15.88**s
7	*CUDA Time lev 1-2*	**3.18**s

Table 3.5: Results of level 2 optimization. Rows indicate the average number of transactions for each setting, the average number of involved prosumers, the average savings, the average savings for energy producers, the average savings for non-producers and the average computational time (respectively without and with the parallel optimization for level 2).

energy production (for example during the summer) are those with the highest number of transactions, involved prosumers and savings. The reader notices that prosumers producing energy will be those benefiting more from level 2 optimization, for which the savings (**Avg Savings (Prod)**) are higher than the average value reported in Table 3.5 for non-producers (**Avg Savings (NProd)**). The reason is that energy transactions need at least an energy producer. Therefore, a producer will usually be involved in a large number of transactions and will therefore save more money, while a non-producer will be, on average, involved in less transactions, if any at all, and will be able to save less money. Furthermore, since in the considered grid the number of energy producers varies between 15 and 41 in a grid with 184 users, this value would be higher if there were more producers involved.

We have also collected results on the relationship between some of the quantities indicated in Table 3.5. In particular, Figure 3.1 describes how the computational time varies depending on the number of transactions occurring in the grid through a 24 hours time interval. Moreover, Figure 3.2 shows how much money is saved through the entire grid in a 24 hours time interval depending on the number of energy transactions occurring within the same time interval. Finally, Figure 3.3 describes how the number of total energy transactions through the grid changes depending on the number of prosumers involved in energy transactions.

Finally, another important result regards the computational time. If parallel optimization is used only for each instance of level 1 optimization preliminary to level 2, the computational time (**CUDA time lev 1**) is on average 15.88s. However, the exploitation of the parallel calculation also for level 2 reduces the computational time (**CUDA time lev 1-2**) by almost five times (3.18s). The reader notices that in configurations where the number of flexible loads and energy producers is higher, the computational time reduction obtained by using parallel calculation for level 2 becomes even higher.

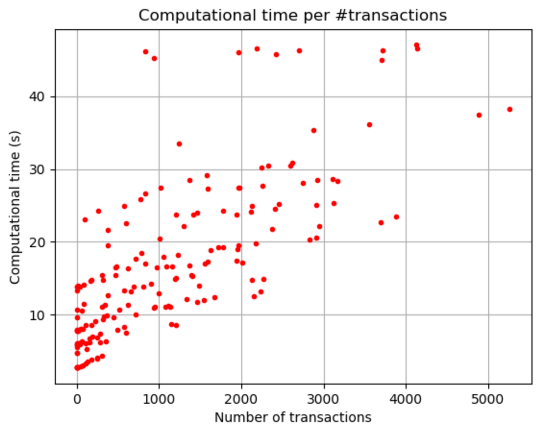


Figure 3.1: Computational time depending on the number of transactions of level 2 optimization. The x axis indicates the number of transactions through a 24 hours interval, the y axis indicates the computational time of the algorithm expressed in seconds.

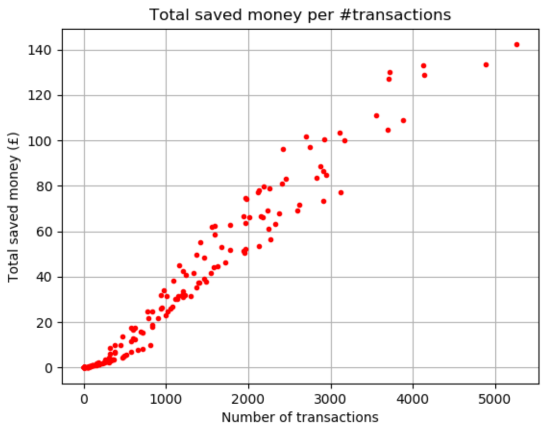


Figure 3.2: Saved money depending on the number of transactions of level 2 optimization. The x axis indicates the number of transactions through a 24 hours interval, the y axis indicates the amount of saved money among all the users expressed in pounds(£).

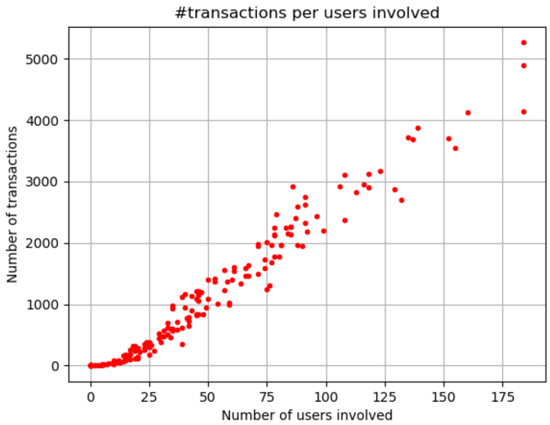


Figure 3.3: Number of transactions in level 2 optimization depending on the number of users involved. The x axis indicates the number of users involved in at least one transaction through a 24 hours interval, the y axis indicates the number of transactions through the grid.

3.6.3 Level 3 Optimization - Automatic

We have run our experiments on all the 180 settings combinations reported in Section 3.4 and the results are shown in Table 3.6. Each row of the table describes:

1. the average peak load after the optimization;
2. the average peak load reduction after to the optimization;
3. the average refund for prosumers in level 3 optimization;
4. the average number of refunded prosumers in level 3 optimization.
5. the average computational time using parallel optimization only for level 1.
6. the average computational time using parallel optimization for levels 1 and 2.

The reader notices that if parallel computation is employed only for level 1, the computational time is on average 23.40s, although this varies depending on the settings such as *Green*/*BAU* or 2020/2030. If parallel computation is employed also for level 2, the average computation time is 11.45s. For comparison, in [51], heuristic algorithms for a problem equivalent to the combination of our level 1 and level 3 were employed for a 50 users grid, and their computation time was 60s or higher. Even with a larger grid, computation time of our algorithm is small enough to make its implementation effective.

		Results
1	Average peak	150.09 kW
2	Peak reduction	18.98 kW
3	Level 3 refund	1.54 £
4	Refunded prosumers	35.92
5	CUDA time lev 1	23.40s
6	CUDA time lev 1-2	11.45s

Table 3.6: Results of level 3 optimization. Refund costs are calculated on a 24 hours time interval.

Chapter 4

Pricing systems: the NRG-X-Change mechanism.

4.1 Context

Nowadays, climate-energy targets and energy policies are pushing global energy systems to undergo a profound energy transformation, namely from fossil and nuclear energy sources to renewables (e.g. wind and solar). On this matter, an important role has been played by the renewable energy support policies implemented in many countries with the aim to promote the installation of renewable energy sources (RES). Such support policies consist on offering long-term contracts to RES producers that compensate the excess of energy that these export to the grid (e.g. by cash payments for unit exported in feed-in tariffs or by free energy in net metering tariffs) in order to make them recover the cost of RES investment. Although such traditional policies have been successful in increasing the penetration of RES (i.e. the share of energy generated from RES is continuously growing, representing 30% in Europe already in 2016[1]), their effect seems to have reached a limit since they are not able to provide any solution to the stability problems that an increment on RES produces to the grid [52, 53]. In more detail, since renewable production is not controllable it can not follow consumption as more traditional energy sources resulting in a mismatch between electricity generation and consumption.

In order to mitigate these issues and increase the renewable hosting capacity of the grid, new incentive mechanisms have been proposed recently in the literature [54, 55, 56, 57]. All of these mechanisms share a common feature: they consider the relation between consumption and production in a given time to set up the incentives and in addition of a payment support function for the producer they also have a second one to incentivise consumers.

Some of these works opted for using a market-based approach [54, 56] in which con-

[1] http://www.eea.europa.eu/data-and-maps/indicators/overview-of-the-electricity-production-2/assessment

sumers and prosumers[2] in a neighbourhood bid in a local market for selling/buying local RES production. Instead, in [55] Mihaylov et al. propose NRG-X-change, a mechanism that combines the advantages of traditional support policies and market-based mechanisms. Similarly to traditional renewable support policies, NRG-X-change does not rely on an energy market - locally produced renewable energy is simply fed into the grid, and is withdrawn by consumers without the need of any complex bidding process. However, unlike traditional support policies, the mechanism offers incentives for both, producers and consumers, linking their incentives at each time slot to a local market-signal - the local energy balance. Flexible loads are the mean through which local energy balance is achieved: therefore, flexibility is what makes this incentive mechanism effective.

The key feature of the *NRG-X-Change* mechanism is exploiting a virtual currency, called *NRGcoin*. The use of this currency can offer some important benefits to energy trading although it does not address some relevant points, such as the local energy trading and the management of congestion. This work aims at exposing how the NRG-X-Change mechanism works relatively to certain aspects, and how it can be modified and improved in order to face the issues we have identified, maintaining the employment of the NRGcoin currency. More in detail, the contributions of this work are:

- We analyzed the NRG-X-Change project and identified its main issues (import and export price functions) and the situations where they appear;
- We designed new import and export price functions that can work better within the NRG-X-Change mechanism and provided theoretical background on those;
- We demonstrated the validity of our proposed functions through mathematical proof, and carried out an experimental evaluation on real data from a grid in Cardiff that measured the efficiency of our proposed functions.

4.2 Background

The NRG-X-Change mechanism has not yet been implemented in actual grids although it has been exhaustively described in different works. In [55] a detailed description of NRG-X-Change has been reported, where authors explained the advantages of using a digital currency and the interactions between grid users, with or without outside parties. Moreover, in [58] a scenario simulating a local grid has been created, showing how NRG-X-Change performs in a realistic environment.

The work in [52] discusses how the usual tariff systems like feed-in tariffs and net metering tariffs present important issues: overconsumption in some periods of the year, overpayment for the energy provider, and underpayment to energy producers. Authors compare a set of state of art support mechanisms in order to solve the above problems. One example of such mechanisms is represented by the auction-based mechanism Nobel [54],

[2]The term has been defined in Chapter 2. In this chapter, and from now on, we will use the definition strictly for energy producers.

which trades energy at a local level based on the stock exchange model. Another one is PowerMatcher [56], based on a market structure regulated by *SD-matchers*, software agents that balance demand and supply in a tree-shaped hierarchical structure. One more work is represented by the negotiation-based mechanism proposed by Capodieci [57], where prosumers are in competition with energy generating companies (*Gencos*) for selling energy, and they negotiate the price with buyers. It is then shown how this mechanism affects the energy price, and also what happens with the implementation of a learning strategy for the buyers. However, all these above-cited works focus on market mechanisms which require final prosumers to get involved into complex bidding processes.

Other works have proposed new tariff schemes, with the main purpose of ensuring grid stability and encouraging active participation of grid users to the energy market. For example, several of these works [59, 60, 61] study from a game-theoretic perspective a type of two-step tariff in which, first, the customer commits to a baseline consumption, and then it is charged based on both their actual consumption and the deviation from the anticipated baseline. In particular, [60, 61] study the properties of this tariff as well as a cooperative game that forms when customers are allowed to aggregate under a group-buying scheme (i.e. behaving like a single, cluster consumer) to collectively reduce their potential deviation. [59] formalized the interaction between the retailer and the customer in such tariffs as a two-player game, studying the optimal strategies for both players as well as the existence of Nash Equilibria. However, notice that the aim of this type of tariff schemes is to make prosumers more predictable to avoid later exchanges in the balancing market. Instead in this work we focus on a renewable support tariff scheme that links the incentives to a local market signal - the local energy balance and/or the local congestion.

Finally, other works focused on mechanisms and contracts for *explicit* demand response in which a subset of consumers is selected and incentivised to reduce their power consumption. [62] formalised and solved this problem as a mechanism design problem while [63] extended this work to include uncertainty about costs. [64] generalises current DR contacts to allow agents to bid how much they want to get paid on for accepting each contract. The mechanism then selects a subset of contracts that minimise the sum of bids, and applies Vickrey-Clarke-Groves prices to pay the agents. Our work takes a different approach focusing instead on *implicit* demand response.

4.3 Background: the NRG-X-Change incentive mechanism

NRG-X-Change is a promising market-based mechanism for trading locally produced renewable energy. It was originally suggested and described in [55]. The mechanism relies on a virtual currency, called NRGcoin. This section outlines the NRG-X-Change mechanism along with its major advantages, providing the definitions of import and export price functions there employed.

4.3.1 NRG-X-Change mechanism

The NRG-X-Change mechanism regulates the payments between the energy producing and consuming agents. It works in time slots of a certain length. In every time slot, the energy production and consumption within the neighborhood is communicated to the local substation. The received information serves as the basis for computing rewards and payments for the corresponding agents. At each time slot, the following steps are applied to each prosumer P and consumer C, agents in the local grid, with the functions f, g and h being always positive for $x, y > 0$.

- A prosumer agent P generates energy E, and, after consuming y of its own energy, feeds the remainder $x = E - y$ to the grid. This information is transmitted to all the nodes in the network, and P generates $f(x)$ NRGcoins, which are awarded to her.

- The substation, which receives the injected energy, transfers $g(x, t_p, t_c)$ NRGcoins to P, where t_p and t_c are, respectively, the total energy produced and the total energy consumed in the community, respectively, at that time slot.

- A consumer agent C pays $h(y, t_p, t_c)$ NRGcoins to the substation for her consumption, where y is the amount of the consumed energy.

Additionally, the NRG-X-Change project considers a local exchange market, where the agents can sell and buy their NRGcoins for the fiat currency[3]. However, the exchange market is outside the scope of our work.

4.3.2 Import and export price functions

The functions f, g and h presented in the Section 4.3.1 regulate the NRGcoin generation and transaction. They were designed to achieve certain objectives. The most important is the balance between t_c and t_p which has to be as close as possible for each time unit in order to give the highest stability to the system. This has to be achieved by encouraging consumers to move their consumption to the time units where energy production is higher.

For these reasons, the functions have been formulated as follows.

- The function f is defined as

$$f(x) = b \cdot x \tag{4.1}$$

  where b is a constant defined by the NRGcoin protocol, determining the number of minted NRGcoins for unit of energy injected into the grid. The generation of NRGcoin has to be small compared to the quantity involved in the transactions: this is achieved by choosing b small enough so that $f \ll g$. The function f is linear, so that the more energy is produced, the more NRGcoins are minted.

[3] legal currency, e.g. Euro, Dollar, etc.

	Export unit price	Import unit price
Underproduction (Overconsumption)	tends to 0	tends to r
Balance ($t_p = t_c$)	q	$\frac{r}{2}$
Overproduction (Underconsumption)	tends to 0	tends to 0

Table 4.1: Unit prices for exporting (e.g. injecting to the grid) and importing (e.g. consuming from the grid) energy in the NRG-X-Change mechanism.

- The function g is defined as

$$g(x, t_p, t_c) = x \cdot \frac{q}{e^{\frac{(t_p - t_c)^2}{a}}} \tag{4.2}$$

where:

  - q is a constant representing the maximum price per unit at which producers are rewarded for their injected energy x (which occurs when the total energy supply matches the total demand); and
  - a is a constant representing the scaling factor for the case when there is no energy balance ($t_p \neq t_c$).

This function is shaped as a bell curve. In [55] the authors described the function similarly so that the maximum payment for the producers is obtained when $t_p = t_c$. Table 4.1 specifies the price ranges in which producers are paid depending on situations of under or over production. Let us observe that as a bell curve the function is symmetrical, meaning that its behavior will be the same if consumption is much higher than the production, and vice-versa. The value of the function ranges from being very close to zero if the difference between t_p and t_c is very high, to q if $t_p = t_c$. From the point of view of the producer this has a bigger impact than the NRGcoin generation, since $f \ll g$.

- The function h is defined as

$$h(y, t_p, t_c) = y \cdot \frac{r \cdot t_c}{t_c + t_p} \tag{4.3}$$

where r is a constant representing the maximum price per unit of consumed energy. As shown in Table 4.1 in the case of overconsumption (i.e. underproduction) this function fixes a unit price which goes from r to $\frac{r}{2}$, where r is achieved in the case of no local production (total overconsumption) and $\frac{r}{2}$ is achieved in the case of local balance. On the other hand, in the case of underconsumption (i.e. overproduction) the unit price goes from $\frac{r}{2}$ to 0, so that the unit price for consumption tends to zero in presence of overproduction. Figure 4.1 shows the energy cost for a consumer, depending on the amount of consumed energy x and considering that the consumer belongs to a local community so that the value of t_c changes according to x. From the point of view of the consumer Eq. 4.3 behaves asymptotically as a linear function, whose angular coefficient goes from 0 to r depending on how high the local consumption is compared to the local production. Note that local consumption depends on y, so the function described in Eq. 4.3 is not actually linear for the consumer.

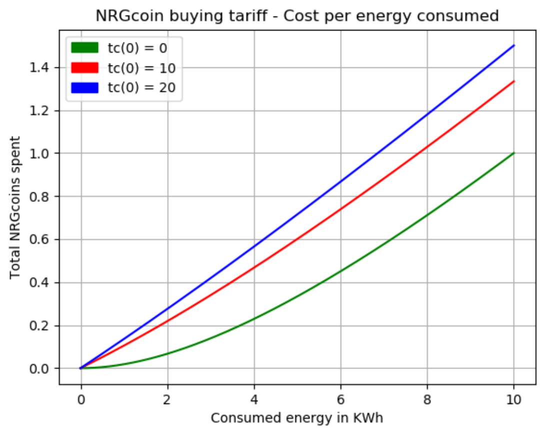


Figure 4.1: The NRG-X-Change cost function: the consumed energy is depicted on the x axis whereas the total cost in NRGcoins is depicted on the y axis. The r parameter is set as 0.2. The different coloured functions correspond to different values of t_c for $x = 0$: the value of t_p is 10 at all times, while the value of t_c for $x = 0$ is 0 for the green function, 10 for the red function and 20 for the blue function.

4.4 Drawbacks of NRG-X-Change mechanism

Although the NRG-X-Change represents an interesting idea, the integration into the real grid calls for a rigorous examination of its properties. In this section, we analyze the NRG-X-Change mechanism identifying potential issues.

One observation relates to the fact that in the original proposal [55] g and h are theoretically defined by the Distribution System Operators (DSO). However, this is in contradiction with the existing legislation because DSO is a public entity responsible for the operation and the maintenance of the distribution system which cannot define the tariffs [65].

Although the original payment functions serve the purpose for many scenarios, several problems can arise when certain conditions occur in the grid. This depends on the specific choices that have been made when formulating these functions.

In fact, in the NRG-X-Change project, the highest priority has been given to the balance between t_p and t_c. As such, the functions regulating NRGcoins transactions between the users and the grid have been defined accordingly.

This brings important consequences. The most important is related to the g function defined as in Eq. 4.2. If a prosumer produces a certain amount of energy, surpassing a certain threshold, it may happen that the more she produces, the less she will receive in return. The reason is that her production breaks the balance between t_p and t_c implying that the prosumer may consider the idea of limiting her own energy production. However, having producers limiting their own energy generation is not something we want from an incentive mechanism.

This is illustrated in Figure 4.2 where the amount of NRGcoins obtained by a prosumer are shown. Parameters of the function have been chosen arbitrarily, but for each choice of them the shape of the function remains unchanged. We can see that the more energy is produced, the more prosumer's revenue increases, until eventually reaching a peak. After that the more energy she injects in the grid, the less profit she will make out of it. Consequently, if her production is too high, it may be more profitable for her to cut it down intentionally.

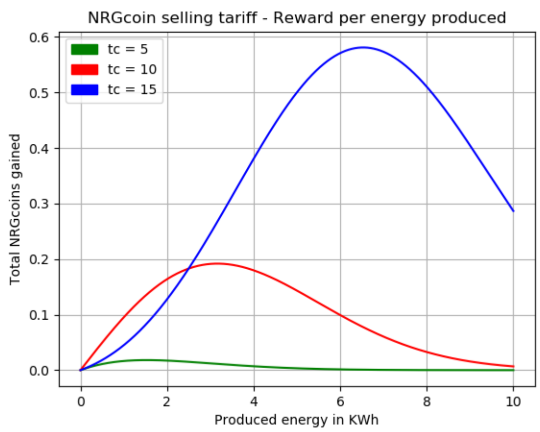


Figure 4.2: Example of a prosumer's revenue for energy produced, with the original g function. The x axis is the amount of energy generated by the prosumer, the y axis is the total amount of NRGcoins gained for her production. The parameters are set as $q = 0.1$, $a = 20$. The different coloured functions correspond to different values of t_c: the value of t_p is 10 for $x = 0$ for each of the functions represented, while the value of t_c is 5 for the green function, 10 for the red function and 15 for the blue function.

Some explanations about the roles of the players involved in the process need to be done. The most important is related to the fact that the DSO cannot impose tariffs [65]. Prices and tariffs are agreed between the users and their suppliers; there is also the aggregator agent, which may let the users interact with the DSO for trading flexibility in exchange of a reward, and allows local grid users to interact for buying or selling energy.

Another important aspect which has to be considered is that the DSO, as operator of the distribution networks, is responsible for the cost-effective and secure transfer of energy over its distribution grid (to and from end users and for the connections to and from the transmission grid) and for ensuring the distribution system's long-term ability to meet electricity distribution demands. In particular, it may happen that the local energy consumption or production exceeds the grid capacity. This condition is referred to as congestion (see also Chapter 2) and it may be harmful for the stability of the grid. For this reason, it is important to discourage local users from consuming too much energy at a given time - or from producing too much energy.

One more important matter which has to be addressed is the fact that prosumers are expected to consume their own energy first, before selling their excess energy. In order for prosumers to consume their own energy first, the buying function h has to always be higher than the selling function g. If this does not happen, a prosumer may actually choose to sell all of her produced energy and buy the energy she needs from another source, since she would have a higher profit. This is something we want to avoid, since it creates unnecessary stress on the grid. In formal terms, the constraint

$$g(x,t_p,t_c) \leq h(x,t_p,t_c) \tag{4.4}$$

has to hold for every value of x, t_p and t_c, taking into account that t_p and t_c depend on x. Thus, when defining the parameters of g and h, they have to be chosen so that Eq. 4.4 is verified. However, this creates a further issue for the functions defined in Eq. 4.2 and Eq. 4.3: for t_c and x very close to zero, it does not exist any combination of the parameters that can satisfy the constraint, as proved in the following proposition.

Proposition 1. *For each possible choice of parameters in Eq. 4.2 and Eq. 4.3, there are values of x, t_p and t_c for which the constraint in Eq. 4.4 does not hold.*

Proof. The constraint, writing explicitly the two functions, becomes:

$$g(x,t_p,t_c) = \frac{q}{e^{\frac{(t_p-t_c)^2}{a}}} \cdot x < \frac{r \cdot t_c}{t_c + t_p} \cdot x = h(x,t_p,t_c). \tag{4.5}$$

Now, we have to take into account that both t_p and t_c depend on x. More precisely, let $\overline{t_p}$ be the value of t_p for $x = 0$, and $\overline{t_c}$ be the value of t_c for $x = 0$. Then,

$$t_p = \overline{t_p} + x$$
$$t_c = \overline{t_c} + x.$$

Substituting in Eq. 4.5, it becomes

$$\frac{q}{e^{\frac{(\overline{t_p}-\overline{t_c})^2}{a}}} < \frac{r \cdot (\overline{t_c} + x)}{\overline{t_c} + \overline{t_p} + 2x}. \tag{4.6}$$

Now, suppose there is a choice for parameters q, a and r for which Eq. 4.6 holds for each possible value of x, $\overline{t_p}$ and $\overline{t_c}$. Let these parameters be $\mathbf{q}$, $\mathbf{a}$ and $\mathbf{r}$, respectively.

We claim that if

$$\begin{aligned} \overline{t_p} &= 1 \\ x+\overline{t_c} &< \frac{\mathbf{q}}{\mathbf{r}} \cdot \frac{1}{e^{\frac{1}{\mathbf{a}}}} \end{aligned} \tag{4.7}$$

Eq. 4.6 does not hold. By substituting in the right hand side member, we obtain

$$\begin{aligned} \frac{\mathbf{r}}{\overline{t_c}+\overline{t_p}+2x} \cdot (\overline{t_c}+x) &< \frac{\mathbf{r}}{1+\overline{t_c}+2x} \cdot \frac{\mathbf{q}}{\mathbf{r}} \cdot \frac{1}{e^{\frac{1}{\mathbf{a}}}} \\ &= \frac{1}{1+\overline{t_c}+2x} \cdot \frac{\mathbf{q}}{e^{\frac{1}{\mathbf{a}}}} \\ &< \frac{\mathbf{q}}{e^{\frac{(1-\overline{t_c})^2}{\mathbf{a}}}} \end{aligned} \tag{4.8}$$

But the last term of the inequality is the left hand side member of Eq. 4.6 after substituting variables according to Eq. 4.7, so we have a contradiction. This means that no matter how we chose the parameters, there are values of x and $\overline{t_c}$ small enough for which the constraint does not hold. □

However, with an appropriate choice of the parameters, the constraint might be satisfied for all the possible values of the variables, except for small values of x and t_c. That is, given a number ε, it is possible to arrange the parameters so that the constraint holds for every value of t_p, and for each x and t_c so that $x+t_c > \varepsilon$, as shown in the following proposition.

Proposition 2. *Let d be the minimum of the function $e^{x^2}-x$ in $\mathbf{R}$. Let g and h be the functions defined in Eq. 4.2 and Eq. 4.3 respectively, and suppose that, given a number $\varepsilon < \frac{d}{2}$, the hypothesis*

$$x+t_c > \varepsilon$$

holds. Then, it is possible to choose the parameters a, q and r in the definitions of g and h so that the constraint defined in Eq. 4.4 holds.

Proof. As shown in the first part of the proof of Proposition 1, it is possible to rewrite the constraint in the form of Eq. 4.6. When $\overline{t_c} > \overline{t_p}$ the inequality is easy to prove (looking at Table 4.1, it is enough to impose $q < 2r$), so from now on we will assume $\overline{t_p} > \overline{t_c}$

Now, our claim is that if $q < \varepsilon \cdot r$, the inequality holds. Substituting it in Eq. 4.6, the constraint becomes

$$\frac{\varepsilon \cdot r}{e^{\frac{(\overline{t_p}-\overline{t_c})^2}{a}}} < \frac{r \cdot (\overline{t_c}+x)}{\overline{t_c}+\overline{t_p}+2x}$$

Which can be strengthened to

$$\begin{aligned} e^{\frac{(\overline{t_p}-\overline{t_c})^2}{a}} &> \varepsilon \cdot \left(\frac{\overline{t_c}+\overline{t_p}+2x}{\overline{t_c}+x}\right) \\ &= \varepsilon \cdot \left(2+\frac{\overline{t_p}-\overline{t_c}}{\overline{t_c}+x}\right) \\ &> \varepsilon \cdot \left(2+\frac{\overline{t_p}-\overline{t_c}}{\varepsilon}\right) = 2\cdot\varepsilon+\overline{t_p}-\overline{t_c}. \end{aligned}$$

We now assume $a = 1$. Calling $u = \overline{t_p} - \overline{t_c}$, the above chain of inequalities reduces to prove

$$e^{u^2} - u > 2 \cdot \varepsilon$$

But the left hand side function is always greater than d, so the inequality holds. Thus, if parameters are chosen as

$$\begin{aligned} a &= 1 \\ q &< \varepsilon \cdot r \end{aligned}$$

the constraint in Eq.4.4 holds. □

It may be of interest to report that the number d defined above is a transcendental number, which can be approximated to 0.7729.

4.5 Proposed payment scheme

This section presents the proposed payment scheme to tackle potential issues identified above.

The first one is the energy production curtailment issue, which is related to the shape of the g function, defined by Eq. 4.2. As shown in Figure 4.2, it may induce prosumers to limit their own energy production, which is something to avoid, except for the specific case where the amount of produced energy causes damage to the grid. In order to avoid curtailment, $g(x,t_p,t_c)$ has to be a monotonic increasing function in x for each possible value of t_p and t_c. This way, the more energy the user produces, the higher her reward will be, encouraging the production of energy.

Also, it has to be considered that, from the prosumer's point of view, t_p depends on x too. This means that, if a prosumer wants to determine her revenue from energy production by using a certain function $g(x,t_p,t_c)$, she must consider that t_p also includes the energy she produces, and t_p will change accordingly to her energy production.

We propose the following formulation of the function g:

$$\begin{aligned} g(x,t_p,t_c) =& P_{max} \cdot \big(g_1(t(x,t_p(x),t_c)) - \\ & g_1(t(0,t_p(0),t_c))\big) - P(x,t_p,t_c) \end{aligned} \tag{4.9}$$

where $P(x,t_p,t_c)$ is a penalty term, which is positive if t_p and t_c are such that a congestion will occur (i.e. $|t_p - t_c| > T$), and zero otherwise, and P_{max} is a scaling factor used to determine the tariff. In this definition, g_1 is a function defined as

$$g_1(t) = \begin{cases} 0 & if t \leq 0 \\ \frac{1}{1+e^{\frac{2t-1}{t^2-t}}} & if t \in (0,1) \\ 1 & if t \geq 1 \end{cases} \tag{4.10}$$

which determines the shape, and t is a function defined as

$$t(x,t_p,t_c) = \frac{t_p(x) - t_c}{2T} + \frac{1}{2} \tag{4.11}$$

where T is a constant, depending on the maximum difference between t_p and t_c which will not correspond to a congestion. Note that we wrote $t_p(x)$ since the local production t_p varies as x varies: for this reason, $t_p(0)$ indicates the value of the local production if the prosumer produces 0.

The formulation of the above function is motivated by the following considerations:

- It is a monotonically increasing function, and as such, it encourages prosumers to produce all the energy they can, avoiding the need for curtailment.

- Its derivative is higher when the difference between t_p and t_c is small, lower when it is large. This means that the balance between consumed and produced energy is encouraged, since the closer t_p and t_c are, the more prosumers will earn in relation to their produced energy.

- Contrarily to other possible functions which address these issues and have these properties, e.g. cumulative distribution functions of continuous random variables, this function does not rely on computation of integrals, which may cause problems in terms of approximation.

- We chose g_1 as a function which is constant outside the $(0,1)$ interval. As a result, our g will not reward prosumers for producing energy in case of congestion; they will instead be penalized by the $P(x,t_p,t_c)$ term in Eq. 4.9. This also encourages curtailment for prosumers if and only if overproduction causes a congestion. Thus, the prosumers, besides being discouraged from producing energy when it is needed less, will actually pay for the energy they inject if it is harmful for the stability of the grid.

- It is possible to prove that the derivative of this function is very low when $t_c \ll t_p$. For this reason, it is also possible to find values for the parameters such that the constraint in Eq. 4.4 is satisfied as shown later on in this section.

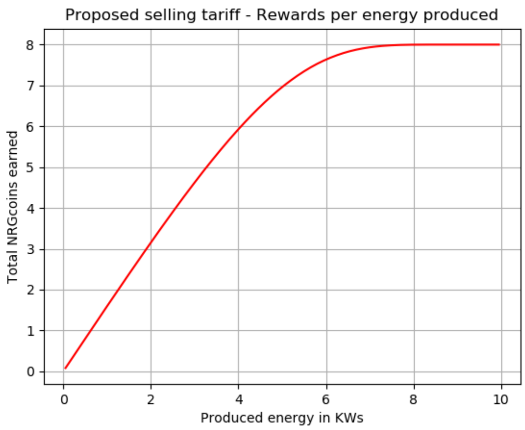


Figure 4.3: Example of a prosumer's revenue for energy produced, with the proposed g function. The x axis represents the amount of energy generated by the prosumer, the y axis represents the total amount of NRGcoins gained for her production. For $x = 0$ we have $t_p = t_c$, while congestion threshold is reached at $x = 10$. Note that for $x > 7$ the tariff is very close to $y = 8$, although this value is only reached for $x = 10$: this can be changed by using different parameters.

In Figure 4.3 we can see the behavior of the proposed function. In particular, when the production gets closer to the congestion threshold, it can be seen that the reward increases much more slowly, but never decreases. This eliminates the need for curtailment for the prosumer.

Example. *Let us suppose, at a certain time unit, the prosumer P produces more energy than she consumes, and the local consumption and production of all the grid users (except her) are perfectly balanced: with the notation we are using, this can be written as* $t_p(0) = t_c = 10$ *kWh. Now, say the parameters are set for example as* $P_{max} = 16$ *NRGcoins and* $T = 10$ *kWh. If at this time unit the prosumer P consumes 2 kWh, the value* t_p *will become* $t_p(2) = 12$ *kWh. The considered prosumer revenue in NRGcoins, according to the proposed tariff, will be, by Eq. 4.9:*

$$g(2,12,10) = 16 \cdot (g_1(t(2,12,10)) - g_1(t(0,10,10))) \tag{4.12}$$

From Eq. 4.11

$$t(2,12,10) = \frac{2}{20} + \frac{1}{2} \quad = \frac{3}{5}$$
$$t(0,10,10) = \frac{0}{20} + \frac{1}{2} \quad = \frac{1}{2}$$

So, substituting in Eq. 4.10, it becomes

$$g_1(t(2,12,10)) = \frac{1}{1+e^{\frac{2\cdot\frac{3}{5}-1}{(\frac{3}{5})^2-\frac{3}{5}}}} = \frac{1}{1+e^{\frac{\frac{1}{5}}{-\frac{6}{25}}}} = \frac{1}{1+e^{-\frac{5}{6}}} \approx 0.697$$

$$g_1(t(0,10,10)) = \frac{1}{1+e^{\frac{2\cdot\frac{1}{2}-1}{(\frac{1}{2})^2-\frac{1}{2}}}} = \frac{1}{1+e^{\frac{0}{-\frac{1}{4}}}} = \frac{1}{1+e^{0}} = 0.5$$

Now, substituting this in Eq. 4.12 we have:

$$g(2,12,10) = 16\cdot(0.697-0.5) = 16\cdot 0.197 \approx 3.152\, NRGcoins.$$

In Figure 4.3 the parameters are set as we just described, and from the graph it can be seen that for $x = 2$ kWh, the value of y is indeed the one we found.

Regarding the function h defined in Eq. 4.3, it does not present particular issues, but some of its aspects can be improved. For example, a function which grows more rapidly when a congestion is about to occur will discourage more easily consumers from using energy in that situation.

A possible function with such a behavior can be defined as

$$h(y,t_c,t_p) = Q_{max}\cdot h_1\left(\frac{tc(y)-tp}{T}+1\right)\cdot y + P(y,t_c,t_p). \tag{4.13}$$

Q_{max} is a parameter used in order to determine the tariff value, corresponding to the maximum cost per unit of energy; T is a parameter corresponding to the congestion threshold like in the definition of Eq. 4.11, and $P(x,t_c,t_p)$ is a penalty factor which is zero for $t_c - t_p < T$ and positive otherwise. Note that we are using the notation $t_c(y)$ because, as it has been pointed out in the definition of g, the value t_c depends on y.

In the above definition, the function h_1 is defined as:

$$h_1(t) = \begin{cases} 0 & \text{if } t \leq 0 \\ \sqrt{t} & \text{if } t \in (0,1] \\ 2-\sqrt{2-t} & \text{if } t \in [1,2) \\ 2 & \text{if } t \geq 2. \end{cases} \tag{4.14}$$

This h function works in a similar way to Eq. 4.3, but has a higher derivative when the local grid is close to the congestion. This means that consumers will be more encouraged to consume energy in case of overproduction since the tariff goes to zero quicker, and, on the other hand, they will be more dissuaded from consuming energy in the case of overconsumption, since the tariff increases much faster when a congestion is close.

In Figure 4.4 this is clearly illustrated. It shows an example of a consumer in a grid context with the following assumptions: for $x = 0$ we have $t_p = t_c$, while the threshold T is equal to 10 kWh, so for $x = 10$ kWh the consumption causes a congestion. In particular, it can be seen that as consumption approaches the congestion, the cost of energy rises faster. This even discourages further grid users from consuming energy at that time, which may lead to a congestion.

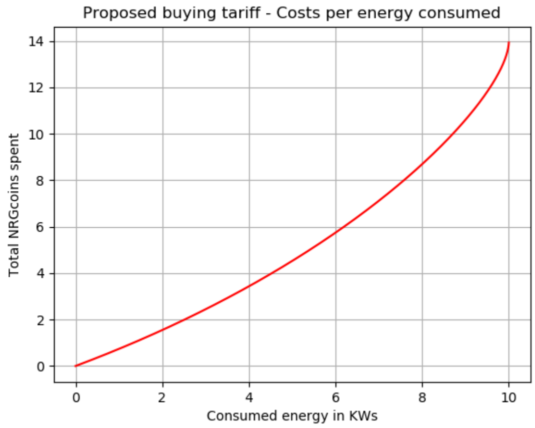


Figure 4.4: Our proposed cost function. On the x axis is the energy consumed, on the y axis is the total cost in NRGcoins; for $x = 0$ we have balance between t_p and t_c, while for $x = 10$ excessive consumption causes a congestion.

Example. *Let C be a consumer which at a certain time unit is about to consume energy without producing any. At that time unit, the total production in the local community is equal to the total consumption of all the users except C. In other words,* $t_p = t_c(0) = 10$ *kWh. Let us suppose also that the parameters are set as* $T = 10$ *kWh,* $Q_{max} = 0.7$ *NRGcoins/kWh. Now, if for example C wants to consume* 6 *kWh in that time unit,* t_c *will become* $t_c(6) = 16$ *kWh. We calculate the energy cost by using the formula in Eq. 4.13. By substituting the parameters, it will become:*

$$h(6,16,10) = 0.7 \cdot h_1(\frac{16-10}{10} + 1) \cdot 6 \tag{4.15}$$

By using the Eq. 4.14 we obtain:

$$h_1(\frac{8}{5}) = 2 - \sqrt{2 - \frac{3}{5}} = 2 - \sqrt{\frac{2}{5}} \approx 1.367$$

Thus, substituting in Eq. 4.15, we obtain:

$$h(6,16,10) = 0.7 \cdot 1.367 \cdot 6 \approx 5.743 \, NRGcoins.$$

In Figure 4.4 the parameters are set as described in this example, and for a consumption of 6 *kWh it can be seen that the cost in NRGcoins is approximately* 5.743.

Furthermore it is possible to verify that the constraint in Eq. 4.4 is verified for the functions g and h we are proposing, as proven in the following proposition.

Proposition 3. *For the functions g and h defined in Eq. 4.9 and Eq. 4.13, it is possible to find values for the parameters such that the constraint defined in Eq. 4.4 holds for every possible value of x,* t_p *and* t_c.

Proof. We first observe that if there is a congestion, the prosumer will have to pay a penalty coming from both functions, and will not receive any income from her selling function. For this reason, we can consider only the cases where $|t_p - t_c| < T$.

Next, we have to take into account the cases when t_p and t_c depend on x. If this occurs, let $\overline{t_p}$ be the value of t_p for $x = 0$, and $\overline{t_c}$ be the value of t_c for $x = 0$. Then,

$$t_p = \overline{t_p} + x$$
$$t_c = \overline{t_c} + x.$$

From now on, we will call

$$u = \frac{T + \overline{t_c} - \overline{t_p}}{2 \cdot T}. \tag{4.16}$$

We now consider two different cases.

Case 1: $u \in [0.1, 1)$. Within this hypothesis, from the definition of h in Eq. 4.13, it results that

$$h(x, t_c(x), t_p) \geq Q_{max} \cdot h_1(2 \cdot u) \cdot x \geq Q_{max} \cdot \sqrt{0.2} \cdot x. \tag{4.17}$$

On the other hand, it is not difficult to see that the derivative of g_1 has an upper bound of 2: from the definition of g (Eq. 4.9), it follows that

$$g(x, t_p(x), t_c) \leq 2 \cdot P_{max} \cdot x. \tag{4.18}$$

Putting Eq. 4.17 and Eq. 4.18 together, we obtain that a sufficient condition for the constraint in Eq. 4.4 to hold is

$$2 \cdot P_{max} \cdot x < Q_{max} \cdot \sqrt{0.2} \cdot x$$

Which is true if and only if

$$P_{max} < \sqrt{0.2} \cdot Q_{max} \tag{4.19}$$

So, for this choice of the parameters P_{max} and Q_{max}, the constraint holds in this case.

Case 2: $u \in (0, 0.1)$. Within this hypothesis, along with Eq. 4.16, we can write h as

$$h(x, t_c(x), t_p) = Q_{max} \cdot \sqrt{2 \cdot u + \frac{x}{T}} \cdot x$$

from which we can deduce

$$h(x, t_c(x), t_p) \geq Q_{max} \cdot \sqrt{2 \cdot u} \cdot x. \tag{4.20}$$

Now we focus on g. Looking back at its definition in Eq. 4.9, and noticing that the t defined in Eq. 4.11 has the following properties:

$$t(x) = 1 - ut(0) = 1 - u - x$$

we can rewrite the equation of g as

$$g(x, t_p(x), t_c) = P_{max} \cdot \left(\frac{1}{1 + e^{\frac{1-2u}{u^2-u}}} - \frac{1}{1 + e^{\frac{1-2u-2x}{(u-x)^2-(u-x)}}} \right). \tag{4.21}$$

Taking the derivative of the g_1 function defined in Eq. 4.10, this implies

$$g(x, t_p(x), t_c) \leq P_{max} \cdot \frac{e^{\frac{1-2u}{u^2-u}}}{\left(1+e^{\frac{1-2u}{u^2-u}}\right)^2} \cdot \frac{2u^2-2u+1}{u^2 \cdot (u-1)^2} \cdot x \tag{4.22}$$

So, putting Eq. 4.20 and Eq. 4.22 together, this implies that a sufficient condition for the constraint in Eq. 4.4 to hold is

$$P_{max} \cdot \frac{e^{\frac{1-2u}{u^2-u}}}{\left(1+e^{\frac{1-2u}{u^2-u}}\right)^2} \cdot \frac{2u^2-2u+1}{u^2 \cdot (u-1)^2} < Q_{max} \cdot \sqrt{2 \cdot u}.$$

Even with $P_{max} = Q_{max}$, for $u \in (0, 0.2)$ the inequality holds. This implies that the condition in Eq. 4.19 is sufficient also in this case, and this concludes the proof. □

The parameters settings satisfying Eq. 4.19 will also satisfy constraint Eq. 4.4 for every possible choice of x, t_p and t_c.

An important remark has to be made. We divided the two cases by imposing

$$u \geq u_0$$

for the first case, and chose $u_0 = 0.1$. The same proof of Proposition 3 can be applied with values of u_0 greater than 0.1. This will give a less strict limitation on the parameters than the one in Eq. 4.19, which becomes

$$P_{max} < M \cdot Q_{max}$$

where M can go up to $\sqrt{0.4}$. However, for the proof to be still valid, the value u_0 has to be chosen so that the inequality in Eq. 4.22 holds for every $u \in (0, u_0)$.

4.6 Simulation Results

In this section we will show how the functions g and h defined by NRG-X-Change and the functions we proposed behave using different combinations of their parameters (Section 4.6.1). We will also experimentally show how much energy is curtailed with the NRG-X-Change g function (Eq. 4.2) depending on the parameter a, while with our g function (Eq. 4.9) there is not such a curtailment (Section 4.6.2). Finally, we will show experimentally how much energy is unnecessarily taken/injected from/into the grid with the NRG-X-Change functions when self-consumption is not adequately encouraged (Section 4.6.3).

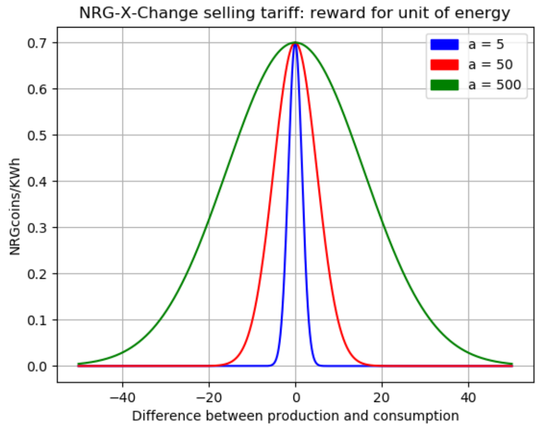


Figure 4.5: Behavior of the NRG-X-Change function g depending on the parameter a. In the x axis there is the value $t_p - t_c$, on the y axis there is the unitary reward in NRGcoins for energy generated. q has been set equal to 0.7 for each example, while a is equal to 5 (blue), 50 (red) and 500 (green), respectively.

4.6.1 Parameter settings

NRG-X-Change

In the original NRG-X-Change mechanism, the function g (defined in Eq. 4.2) depends on two parameters, q and a. The former determines the maximum value of the tariff for unit of produced energy, while the latter determines how quickly the reward decreases as the difference between t_p and t_c increases. This is illustrated in Figure 4.5 where it can be seen that the unitary reward is higher when $t_p = t_c$, and it decreases when t_p and t_c become different at a rate depending on a. The h function, on the other hand, scales linearly with the parameter r.

Proposed functions

Our proposal for the function g (defined in Eq. 4.9) depends on two parameters, T and P_{max}. The behavior of P_{max} is similar to q in the function defined in the NRG-X-Change project, and defines the magnitude of the unitary reward. T is the threshold of the difference between t_p and t_c at which congestion occurs, and depending on it the maximum reward can increase and the accepted difference between t_p and t_c can decrease, or vice-versa, as shown in Figure 4.6. For this reason, the maximum reward is not determined by the parameter P_{max} alone, but also by the choice of T. The function h depends on the parameters Q_{max}, which behaves exactly like P_{max} in g, and T; its dependence on these parameters is the same as g. To note that g is the function defined in Eq. 4.9 whereas h is defined in Eq. 4.13.

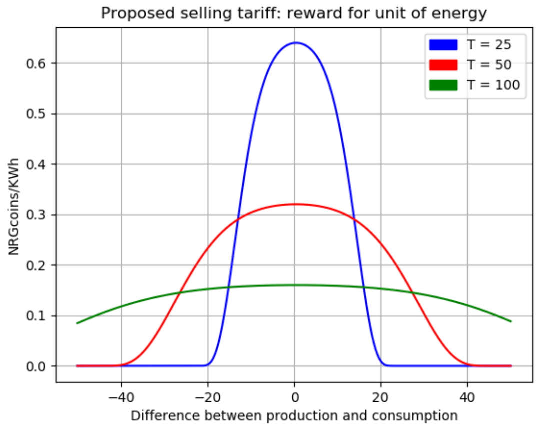


Figure 4.6: Behavior of our proposed function g depending on the parameter T. In the x axis there is the value $t_p - t_c$, on the y axis there is the unitary reward in NRGcoins for energy generated. P_{max} has been set equal to 16 for each example, while T is equal to 25 (blue), 50 (red) and 100 (green) respectively.

4.6.2 Energy production curtailment

Simulations have been run to measure the amount of energy curtailment with the NRG-X-Change selling function. For the simulation, real data from a grid in Cardiff have been used: more precisely, the data described in Section 3.4, with the *Green*, 2030 and *High* settings. The grid contains 184 users, 41 of which are prosumers.

For the simulations, scenarios of local grids have been created. From now on, in this section, we will refer to a *local grid* as a grid made of 40 users, chosen randomly from the 184 users of Cardiff, so that 10 of them are prosumers. We simulated the behavior of the users in real settings where they choose the actions that will maximize their economic profit: they will curtail energy if and only if it would be profitable for them, and the amount of energy curtailed will be chosen in order to get the highest possible profit for them. We simulated 400 different local grids, and measured:

- the number of times producers curtail energy production, during 24 hours, considering time intervals of 15 minutes **(# Curtailments)**;
- how much energy has been curtailed among all the prosumers during a 24 hours time interval **(Energy curtailed)**;
- the average quantity of energy production reduction for each production curtailment. In other words, the ratio between the second and the first quantities above measured. **(Ratio)**

This process has been carried out employing the selling function of the NRG-X-Change mechanism (the one in Eq. 4.2). In Table 4.2 the results relative to the NRG-X-Change selling functions are reported.

Value of a	**# Curtailments**	**Energy curtailed**	**Ratio**
1	71.02	3.74 kWh	0.053
2	78.99	3.83 kWh	0.049
3	80.09	3.73 kWh	0.047
5	74.09	3.06 kWh	0.041
10	42.06	1.24 kWh	0.029
15	17.99	0.41 kWh	0.023
20	5.98	0.12 kWh	0.02
25	0.26	0.005 kWh	0.019
30	0.03	¡0.025 kWh	0.016

Table 4.2: Curtailment in a local grid with 40 users, 10 of which are prosumers. The columns indicate, in order: value of parameter a in Eq. 4.2, instances of energy production curtailment in a 24 hours time interval, total curtailment in the local grid in a 24 hours time interval, and the quantity of energy curtailed for each instance of energy production curtailment.

The amount of curtailed energy depends on the parameter a in Eq. 4.2. From Table 4.2, it can be seen that for relatively low values of a there is an energy curtailment on average around 3.8 kWh, while without curtailment the average production in 24 hours is 89.75 kWh. Also, for low values of a, the number of times a producer applies curtailment goes up to 80. The reader notices that the average daily electricity consumption for the Cardiff grid users is around 14.11 KWh. As the parameter a becomes higher, the number of instances of curtailment becomes larger on average, although in several scenarios this does not occur. The amount of curtailed energy for each instance, on the other hand, becomes smaller when a becomes larger.

The same simulation has been run with the function we proposed in Eq. 4.9. Unlike the NRG-X-Change mechanism, no prosumer was encouraged to curtail their production. This is because our proposed scheme is designed to promote energy balance without curtailment. Curtailment is only considered in the extreme case of a congestion caused by overproduction. These results show that the proposed scheme allows to avoid significantly the curtailment of energy, which is working towards the energy policy targets to increase the renewable hosting capacity of the grid.

4.6.3 Self-consumption

As stated in Proposition 1 that, once the parameters for the selling function g and for the buying function h proposed by NRG-X-Change have been chosen, there are always values for x, t_p and t_c such that the equation

$$g(x,t_p,t_c) < h(x,t_p,t_c)$$

does not hold. For this reason, prosumers may be in some cases encouraged to buy energy from the utility to satisfy their own consumption while injecting the produced energy into the grid. This situation contradicts to the fact that the produced energy should be consumed locally thereby reducing stress on the grid and losses due to transmission.

It can be easily deduced by looking at the functions that the likelihood of an intentionally reduction of the self-consumption depends mainly on the choice of parameters for g and h, and on the number of users/prosumers of the grid, since the higher the number of prosumers/consumers, the higher t_p and t_c. More in detail, from the definitions of g and h, it can be seen that an intentionally reduction of the self-consumption is more likely to occur with higher values of q and a, and with lower values of r. However, q and r directly determine the maximum value for the tariffs, so once they are chosen, the chance of a reduction on the self-consumption depends mainly on the parameter a, as we have shown in Proposition 2. In fact, lowering this parameter makes this situation less likely to occur, and it can be avoided if a certain value for $x+t_c$ is guaranteed at any time; however, if the grid is smaller (and in particular, in the case with only one producer) there may be the chance of a reduction of the self-consumption. Also, a low value for a reduces the reward for the produced energy when t_p and t_c do not match perfectly, so there is a limit on how much it can be reduced.

In order to illustrate this issue we carried out an analysis defining and running simulations on local grids, defined as in Section 4.6.2. Users of the grid had the possibility to see their costs/revenues for energy in case of self-consumption (of all their needed energy), and to reduce this amount (therefore buying additional energy from the grid, and selling more energy) if it was economically more profitable. We simulated 500 small local grids: they have been built from the Cardiff grid data using a similar procedure already introduced in Section 4.6.2. For each small local grid we considered a period of 24 hours (00:00 - 23:45) and time intervals of 15 minutes, and measured the following quantities:

- The number of times an intentional reduction of the self-consumption occurred in the local grid through the 24 hours period, considering time intervals of 15 minutes. It corresponds to the number of time slots in which some reduction in the self-consumption occurs **(# Self-consumption reductions)**.
- The total amount of energy which has not been self-consumed in all the grids during 24 hours **(Energy not self-consumed)**.

The Cardiff grid we have used for simulations had ToU tariffs: we chose the parameters for g and h such that their values in NRGcoin are as close as possible to the values of the ToU tariffs in fiat currency: since the NRG-X-Change tariffs depend on various factors while ToU tariffs are fixed, we chose the parameters so that, on average, the values of g and h are as close as possible to the Cardiff grid ToU tariffs. In our case, with the appropriate unit of measurement, q may range between 0.06 and 0.1, and r between 0.2 and 0.3. We run simulations with different values of a, ranging between 100 and 200: reductions on the self-consumption occurred only when $q = 0.1, r = 0.2$. Results can be

Value of a	**# Self-consumption reductions**	**Energy not self-consumed**
100	0.004	0.0009 kWh
125	0.454	0.1186 kWh
150	2.634	0.701 kWh
175	5.504	1.5279 kWh
200	7.118	2.1149 kWh

Table 4.3: Reduction of the self-consumption in a local grid with 40 users, 10 of which are prosumers. The columns indicate, in order: value of parameter a in Eq. 4.2, instances of reduction on the self-consumption in a 24 hours time interval, and total amount of energy unnecessarily taken from/injected into the grid.

seen in Table 4.3. In particular, with these settings, it can be seen that the average number of intentional reductions of the self-consumption in a local grid can go up to 7.118 kWh. Also, we recall that a reduction on the self-consumption occurs when a prosumer does not self-consume all of her produced energy: as a result, a certain amount of energy is unnecessarily withdrawn from the grid and injected into the grid. In our simulations, the total energy through the grid goes up to 2.1149 kWh in a 24 hours time interval. This causes additional stress to the grid and potential additional problems for the DSO, and can be avoided by ensuring the existence of a set of parameters that prevents such reductions on the self-consumption to occur. In Proposition 3 we defined conditions for the parameters to ensure this.

Chapter 5

A Game Theory focus on the NRG-X-Change mechanism

5.1 Context

The work described in this chapter is a direct consequence of what has been done in Chapter 4. The context in which we are working in, from a smart grid perspective, is the exact same that has been described in Section 4.1. In particular, this part starts again from the idea of the mechanism *NRG-X-Change*, which has been described exhaustively in [55] and in Section 4.3. The purpose of this chapter is to create a new incentive mechanism based on NRG-X-Change and the use of *NRGcoin* using as a baseline the works in [55] and Chapter 4 [2], but further improving the critical points of the mechanism and studying its behavior in a context with selfish agents. The contributions of the work this chapter are therefore the following:

- We analyze the NRG-X-Change project and identify its main issues (import and export price functions) and the situations where they appear;
- We design new import and export price functions that can work better within the NRG-X-Change mechanism by exploiting known results in game theory about Nash equilibria, and provide theoretical background on those;
- We demonstrate the validity of our proposed functions through mathematical proof, and carry out an experimental evaluation on real data from a grid in Cardiff that measured the efficiency of our proposed functions.

Flexibility is a key concept for the work in this chapter too. Like in the previous chapter, its usage is what allows local energy balance to be realized, and therefore is what motivates the existence itself of the selling and buying functions. However, in this case, flexibility is also necessary for defining the game theory context, which is the primary focus of this work.

5.2 Background

For what incentive mechanisms are concerned, literature is quite extensive, and it has been described exhaustively in Section 4.2. In particular, several mechanisms similar to NRG-X-Change [52, 54, 56, 57] have been described and reported in that section.

Literature on employment of game theory for the smart grid context is extensive. [66] present a survey which describes comprehensively how game theory methods have been used for smart grid. The work done by [67] is probably one of the closest to the work we propose here in terms of employment of game theory, although its focus is mainly devoted to manage demand response whereas ours is on the incentive mechanism. The game theory usage done by [68] and [69] has also been of inspiration for us, since the model we will propose in this chapter has been heavily influenced by their game theory modeling in the smart grid context. The work from [70] contains another survey describing the use of game theory for smart grid, although it focuses on collaborative games, whereas our work describes mainly a situation where each agent is selfish.

5.3 Modelization of the problem

In this section the modelization of the problem will be discussed. We built a model from a game theory perspective in order to explain the effect of the incentive mechanism on the agents. Our construction was inspired by [68] and [69].

5.3.1 Game Theory approach to our Smart Grid problems

First, we introduce our notation for game theory, following what has been defined in Chapter 2. We define a game $G = (U, S, Q)$ in the following way:

- $U = \{U_1, \ldots, U_N\}$ is the set of players.
- $S = \{S_1, \ldots, S_N\}$, where for any $i \in \{1, \ldots, N\}$, S_i is the set of player U_i's strategies.
- $Q = \{q_1, \ldots, q_N\}$, where for any $i \in \{1, \ldots, N\}$, $q_i : \prod_{j=1}^{N} S_j \to \mathbf{R}$ is the payoff function for player U_i.

Let us consider a grid with N users: we will now define a game on it. The players are the grid users, which will thus be denoted as U_i, for $i \in \{1, \ldots, N\}$. In what follows we describe the users' payoff functions and the strategies they are allowed to choose.

For each user U_i, we define a vector $\mathbf{c_i}$ which indicates the energy *consumption* of U_i, and a vector $\mathbf{p_i}$ which indicates the energy *production* of U_i. The size of these vectors is the number of intervals in which the considered time horizon is divided: for example, in a scenario with a 24 hour time horizon with 15 minutes intervals, the size of $\mathbf{c_i}$ and $\mathbf{p_i}$ is 96. Each component of both $\mathbf{c_i}$ and $\mathbf{p_i}$ is a non-negative real number. We will denote this number of time intervals as T.

One important consideration has to be made about the values of $\mathbf{c_i}$ and $\mathbf{p_i}$: in this chapter we assume that each user self-consumes all the energy she needs before exporting it or imports the extra energy she needs, and the values of $\mathbf{c_i}$ and $\mathbf{p_i}$ are considered after self-consumption is taken into account. This, in formal terms, denoting as $\mathbf{c_i}(t)$ the $t-$th element of $\mathbf{c_i}$, is written as

$$\mathbf{c_i}(t) \cdot \mathbf{p_i}(t) = 0 \;\; \forall t \in 1, \dots, T \tag{5.1}$$

for each $i \in \{1, \dots, N\}$. Of course, if U_i is a consumer which does not produce energy, $\mathbf{p_i}$ is the zero vector. Later in this chapter, we will ensure that all the payment functions we propose actively encourage energy producers to self-consume their produced energy.

Now, we can define the utility function q_i for each grid user as the sum of the utility in the different time intervals, namely:

$$q_i = \sum_{t=1}^{T} q_i(t) \tag{5.2}$$

where the utility of a user U_i at time interval t is defined as:

$$q_i(t) = g(\mathbf{p_i}(t), t_p, t_c) - h(\mathbf{c_i}(t), t_p, t_c). \tag{5.3}$$

where g and h, with the notation of Chapter 4, represent respectively the reward for the energy produced and the cost for the energy consumed are two pre-determined functions, and have the property that $g(0, a, b) = h(0, a, b) = 0$ for every $a, b \in \mathbf{R}$. Fixed a time unit t, the aforementioned quantities t_p and t_c are defined as

$$\begin{aligned} t_p &= \sum_{i=1}^{N} \mathbf{p_i}(t) \\ t_c &= \sum_{i=1}^{N} \mathbf{c_i}(t) \end{aligned} \tag{5.4}$$

and represent the total production and the total consumption through the grid at that given time. In particular, at a given time t, t_p depends on every $\mathbf{p_i}(t)$ and t_c on every $\mathbf{c_i}(t)$.

Next, we show the possible strategies for the players. More precisely, how the loads can be modeled in our scheme. An example of this has been seen in the work from [69]. In the following we will indicate the loads that we have taken into consideration for our work.

- **Production**: This is $\mathbf{p_i}$. It represents the energy production of the user U_i, and it is a fixed vector unless the energy production is somehow modulable, which is not our case.

- **Fixed consumption**: This is a fixed vector $\mathbf{f_i}$. It represents the fixed part of the energy consumption of the user U_i. In this chapter it will be treated as a known vector, although it has to be forecast and may present uncertainty: a possible approach to this issue can be the one shown in Section 3.5.2, which treat the components of the vector as probability distributions.

- **Shiftable load**: This is a vector $\mathbf{h_i}^j$, which represents a consumption load of user U_i which can be shifted in time. The user can control how many time units this vector can be shifted by, although sometimes they have restrictions on this. Given the vector $\mathbf{h_i}^j$, the notation $r_k(\mathbf{h_i}^j)$ will indicate that vector shifted by k places. In other words, r_k indicates the rotation by k places: if $k = 0$ there is no rotation, therefore $r_0(\mathbf{h_i}^j) = \mathbf{h_i}^j$.

Thus, if a user has n shiftable loads $\{\mathbf{h_i}^1, \dots, \mathbf{h_i}^n\}$, and decides to shift each $\mathbf{h_i}^j$ by a quantity k_j, we have

$$\mathbf{c_i} = \mathbf{f_i} + \sum_{j=1}^{n} r_{k_j}(\mathbf{h_i}^j). \tag{5.5}$$

The set of possible strategies of U_i is then the set of all her possible vectors $\mathbf{p_i}$ and $\mathbf{c_i}$, depending on how the loads are shifted.

5.4 Peer-to-peer market design

This section presents the proposed peer-to-peer market for energy trading.

5.4.1 Purpose

The goal of this chapter is to modelize an incentive mechanism for renewable energy, using as a baseline the NRG-X-Change [55] mechanism, but improving some of its critical points. More precisely, these points are:

1. Ensuring that prosumers are not incentivized to curtail their own energy production, unless this is necessary for the grid stability.

2. Taking into account the possibility of a congestion, and actively attempting to prevent it.

3. Encouraging prosumers to self-consume the energy they produce before selling, or before buying from the grid the excess energy they need.

4. Analyzing the behavior of the users if they modulate selfishly their production and consumption, and ensure that they reach an equilibrium.

The first three points have been addressed in Chapter 4, by creating functions which describe rewards for energy production and costs for energy consumption respectively. The scope of this chapter is to create new functions which address all of the four points mentioned above.

5.4.2 The NRG-X-Change Incentive Mechanism

The NRG-X-Change mechanism has already been described within detail in Section 4.3. In particular, the mechanism's functioning is determined mainly on two functions: one, named g, which describes the reward for the energy injected into the grid by prosumers, and the other, named h, which describes the cost for the energy taken from the grid by consumers. The original NRG-X-Change reward and cost functions have been described in Eq. 4.2 and Eq. 4.3 respectively.

In order to face the first three aforementioned issues, in Chapter 4 we created one new reward function and one new cost function, which have been defined respectively in Eq. 4.9 and Eq. 4.13. These new functions solve many of the issues of the original NRG-X-Change work, but still have to be analyzed from a game theoretical perspective. In order to do so, we have to use the model defined in Section 5.3 and analyze the behavior of the players with the payoff functions relative to each system.

For the original NRG-X-Change system, it is not difficult to write $q_i(t)$: using Eq. 5.3, it can be written as

$$\begin{aligned} q_i(t) &= x \cdot \frac{q}{e^{\frac{(t_p - t_c)^2}{a}}} \quad \text{if} \quad y = 0 \\ q_i(t) &= -y \cdot \frac{r \cdot t_c}{t_c + t_p} \quad \text{if} \quad x = 0 \end{aligned} \tag{5.6}$$

keeping in mind that either x or y is equal to zero, as seen in Eq. 5.1.

For the system realized in Chapter 4 the expression for q_i is more complicated to write, although it may be simplified if we consider different cases. Since at least one between x and y is equal to zero, substituting Eq. (4.10) and (4.11) in Eq. (4.9) and Eq. (4.14) in Eq. (4.13):

- **If** $y = 0$, i.e. the user's production is greater than the user's consumption, we have

$$q_i(t) = \frac{P_{max}}{1 + e^{\frac{2^{\frac{t_p^{-i} + x - t_c^{-i}}{B}}}{\left(\frac{t_p^{-i} + x - t_c^{-i}}{B}\right)^2 - \frac{1}{4}}}} - \frac{P_{max}}{1 + e^{\frac{2^{\frac{t_p^{-i} - t_c^{-i}}{B}}}{\left(\frac{t_p^{-i} - t_c^{-i}}{B}\right)^2 - \frac{1}{4}}}} \tag{5.7}$$

- **If** $x = 0$, i.e. the user's consumption is greater than the user's production, we have
  - If $t_p^{-i} > t_c^{-i}$,

$$q_i(t) = -y \cdot Q_{max} \cdot \sqrt{\frac{t_c^{-i} + y - t_p^{-i}}{B} + 1} \tag{5.8}$$

  - If $t_p^{-i} < t_c^{-i} + y$,

$$q_i(t) = -y \cdot Q_{max} \cdot \left(2 - \sqrt{1 - \frac{t_c^{-i} + y - t_p^{-i}}{B}}\right) \tag{5.9}$$

An important remark has to be made: since the production suffers from no curtailment, but x is the value of production minus consumption, a prosumer may change this value by changing her consumption.

5.4.3 New proposals

Among the points listed in Section 5.4.1, the fourth is the one which still has to be solved. Basically we want to check whether a Nash equilibrium (NE) exists for the previously defined incentive systems, and in case it does not, to create new possible selling and buying functions which guarantee the existence of a NE while fulfilling the first three points listed in Section 5.4.1. The concept of Nash equilibrium has been described in Chapter 2.

For the first two systems, our strategy has been the following: first, create a simulation of the described game in order to see empirically whether a NE exists or not, and if the answer was positive, find a mathematical proof of its existence. As we will see in Section 5.5, these functions do not guarantee the existence of a NE.

We wanted to create selling and buying functions which guarantee the existence of a NE in the game we described. For this, we exploited a well-known result: a game with a concave payoff function admits a pure NE [68]. So, if we can create a concave selling function and a convex buying function, the utility function defined in Eq. 5.3 will be concave and the theorem proven by [71] will guarantee the existence of a pure NE.

However, this may create conflicts with the other points, in particular with the condition for avoiding curtailment and the self-consumption condition. For the former, it is enough to ensure that the selling function g is a monotonic function with respect to x. In order to check the latter, however, we want to check that the condition described in Eq. 4.4 holds for each $x > 0$ and for each $t_p, t_c \geq 0$.

Some ideas for functions that respect the above constraints may be the following. We will denote

$$Z = t_p^{-i} - t_c^{-i} + B. \tag{5.10}$$

- For the selling function g:
  - **[Logarithm]** One idea may be to base the function on logarithm. In this case, the function can be

    $$g(x, t_p, t_c) = k_1 \cdot \left(ln \frac{x + Z + a_1}{Z + a_1} \right) \tag{5.11}$$

    where a_1 and k_1 are parameters, and $a_1 > 0$.
  - **[Square root]** Another idea may be to use the square root function as a base. In this case, a possible candidate is

    $$g(x, t_p, t_c) = k_1 \cdot (\sqrt{x + Z + a_1} - \sqrt{Z + a_1}) \tag{5.12}$$

    where a_1 and k_1 are parameters, and $a_1 > 0$.

- For the buying function h:
  - **[Quadratic]** One possibility is to use the quadratic function. In this case, a candidate may be

$$h(y,t_p,t_c) = k_2 \cdot ((y - Z + a_2)^2 - (a_2 - Z)^2) \tag{5.13}$$

    where a_2 and k_2 are parameters, and $a_2 > 2B$.
  - **[Square root]** Another possibility for this may be the negative square root function. In this case, our function may have the following form:

$$h(y,t_p,t_c) = k_2 \cdot (\sqrt{Z + a_2} - \sqrt{Z + a_2 - y}) \tag{5.14}$$

    where a_2 and k_2 are parameters, and $a_2 > 0$.

To each of the proposed functions a term P has to be added. For selling functions, P is a function determined by x, t_p and t_c such that its value is zero unless $t_p - t_c > B$, in which case it is negative. For buying functions, P depends on y, t_p and t_c and its value is zero unless $t_c - t_p > B$, in which case it is positive. In other words, P is a penalty term which punishes overproduction/overconsumption when it causes a congestion. Notice that, with the used notation, a congestion occurs if and only if $Z \notin (0, 2B)$.

These functions have been designed to have a certain behavior in relation to the variable Z. When local energy consumption is higher than local energy production, the energy generated by producers is valuable since it helps covering the consumption needs of the community: in this case the selling functions assume higher values, and consequently producers are paid more for the energy they produce. On the other hand, when local energy production is higher than local energy consumption, the surplus energy injected by the producers is not needed by the consumers, so the selling functions assume lower values and energy producers are paid less. As far as the buying functions are concerned, when local energy consumption is higher than local energy production we want to discourage consumers from consuming energy, since energy has to be bought from outside the community: for this reason, buying functions assume higher values and the price for buying energy is higher for consumers. Conversely, when local energy consumption is lower than local energy production, we want to encourage consumers to consume energy, since they would use the energy generated by the producers: for this reason, buying functions assume lower values and consequently energy costs less for the consumers.

It is not difficult to check that the proposed selling functions are monotonic: the more energy a user produces, the more she is paid for. For this reason the user is discouraged from curtailing her own energy production, since curtailment would decrease her profits. The only exception is the case of congestion for overproduction, where the penalty term discourages producers from generating energy by lowering their profits. This is the first condition described at the beginning of Section 5.4.1, which is then respected by our functions: we want to ensure that the functions we create satisfy all of them. The second condition that has to be fulfilled is the fact that both selling and buying functions actively consider the possibility of congestion: since the functions we are creating have a penalty term for congestion, they fulfill this condition. We will prove in Section 5.5.1 that the proposed functions respect also the third condition, while the last condition holds because of the theorem proven in [71]. This theorem implies that our game guarantees the existence of a Nash equilibrium if the payoff functions are concave; it is easy to check that this is true from Eq. 5.3, since the proposed selling functions are concave and the proposed buying functions are convex.

5.5 Simulations and results

This section is divided in two parts. In Section 5.5.1 the constraint for encouraging self-consumption due to pricing is proven to hold for the newly proposed functions, while in Section 5.5.2 it is proven that the baseline functions do not guarantee the existence of a NE, while the new proposed functions always reach a NE as expected from the theoretical results.

5.5.1 Parameter settings

The goal of this section is to verify that the functions proposed in this work respect the self-consumption condition, which is expressed in Eq. 4.4. We will analyze the behavior of the functions, and check if there is a choice for the parameters such that the condition is fulfilled.

Candidate 1: logarithm selling, quadratic buying

This is the case where the selling function g is the one described in Eq. 5.11, while the proposed buying function h is the one described in Eq. 5.13.

We want to check that Eq. 4.4 holds for some choice of the parameters of these two functions. By rewriting this condition with the two functions, we want to prove that there is a choice for the parameters k_1, a_1, k_2 and a_2 such that

$$k_1 \cdot \left(\ln \frac{x+Z+a_1}{Z+a_1}\right) < k_2 \cdot \left((x-Z+a_2)^2-(a_2-Z)^2\right)$$

holds for each $x>0$ and $Z \in (0,2B)$.

Lemma 1. *For each $x>0$, $Z>0$, $a_1 \geq 1$ and $a_2 \geq 2B+1$, and choosing $k_1 = k_2 > 0$, the inequality*

$$\ln\left(1+\frac{x}{Z+a_1}\right)^{k_1} < k_2 \cdot x \cdot (x-2Z+2a_2) \tag{5.15}$$

holds.

Proof. Considering the exponent, Eq. 5.15 becomes

$$\left(1+\frac{x}{Z+a_1}\right)^{k_1} < e^{k_2\cdot x\cdot(x-2Z+2a_2)}$$

Since $k_1 = k_2$, we can remove the equal exponents and the inequality that we want to prove becomes

$$1+\frac{x}{Z+a_1} < e^{x\cdot(x-2Z+2a_2)}$$

Thus, exploiting the properties $a_1 \geq 1$ and $a_2 \geq 2B+1$, the chain of inequalities

$$1+\frac{x}{Z+a_1} \leq 1+\frac{x}{Z+1} < 1+x < e^{x^2+x} \leq e^{x\cdot(x-2Z+2a_2)}$$

completes the proof, since it holds for every value of $x > 0$ and $Z \in (0, 2B)$. □

Proposition 4. *The function defined in Eq. 5.11 with $a_1 = 1$, together with the function defined in Eq. 5.13 with $a_2 = 2B+1$, satisfies the self-consumption condition defined in Eq. 4.4 if $k_1 = k_2 > 0$.*

Proof. Thanks to Lemma 1, with the aforementioned choice of parameters, the chain of inequalities

$$\begin{aligned} & k_1 \cdot \ln\left(\frac{x+Z+a_1}{Z+a_1}\right) = \ln\left(\frac{x+Z+a_1}{Z+a_1}\right)^{k_1} = \\ & = \ln\left(1+\frac{x}{Z+a_1}\right)^{k_1} < k_2\cdot x\cdot(x-2Z+2a_2) = \\ & k_2\cdot\left((x-Z+a_2)^2-(a_2-Z)^2\right). \end{aligned}$$

holds for each $x > 0$, $Z \in (0, 2B)$. □

Candidate 2: square root selling, negative square root buying.

This is the case where the selling function g is the one described in Eq. 5.12, while the proposed buying function h is the one described in Eq. 5.14.

Proposition 5. *There is a choice for the parameters k_1 and a_1 for the selling function defined in Eq. 5.12, and for the parameters k_2 and a_2 for the buying function defined in Eq. 5.14, such that these two functions fulfill the condition defined in Eq. 4.4. More specifically, this happens if $k_1 = k_2 > 0$ and if $a_1 \geq a_2 + 2B$.*

Proof. The self-consumption condition can be written as

$$\begin{aligned} & k_1 \cdot \left(\sqrt{x+Z+a_1} - \sqrt{Z+a_1}\right) < \\ & k_2 \cdot \left(\sqrt{Z+a_2} - \sqrt{Z+a_2-x}\right). \end{aligned}$$

Substituting $k_1 = k_2$ and canceling them out, it becomes

$$\sqrt{x+Z+a_1}-\sqrt{Z+a_1}<\sqrt{Z+a_2}-\sqrt{Z+a_2-x}.$$

Rationalizing the denominators, this holds if and only if

$$\frac{x}{\sqrt{x+Z+a_1}+\sqrt{Z+a_1}}<\frac{x}{\sqrt{Z+a_2}+\sqrt{Z+a_2-x}}$$

and considering denominators, since we want to prove it for $x > 0$, this is true if and only if

$$\sqrt{x+Z+a_1}+\sqrt{Z+a_1}>\sqrt{Z+a_2}+\sqrt{Z+a_2-x}.$$

Now, if $a_1 \geq 2B + a_2$, we have the chain of inequalities

$$\begin{gathered}\sqrt{x+Z+a_1}+\sqrt{Z+a_1}>2\sqrt{a_1}\geq\\ 2\sqrt{2B+a_2}>\sqrt{Z+a_2}+\sqrt{Z+a_2-x}\end{gathered}$$

which holds for every $x > 0$, $Z \in (0, 2B)$, and this completes the proof. □

5.5.2 Results

A script in *Python* language has been created in order to simulate this framework. For this, real data relative to a Cardiff grid with 184 users, 41 of which are prosumers, have been used. Data have already been described exhaustively in Section 3.4.

In this section, we show two important facts. First, the NRG-X-Change incentive system and the one described in Chapter 4 do not guarantee the existence of a NE in the game described in Section 5.3. Second, we will see that a NE always exists with the functions we have defined in the end of Section 5.4, as expected by the known theoretical results.This is how the game has been simulated:

1. A set of N users, $U_1, \ldots, U_N$, is randomly chosen from the 184 Cardiff users: depending on the scenario that we want to simulate, N is pre-determined, and the number of users that are prosumers is pre-determined as well. The predicted consumption and production of each user U_i is known. Each user has exactly one shiftable load, which is randomly allocated at the beginning.

2. This step is repeated sequentially for each user U_i, i.e. starting from U_1, then U_2, and arriving to U_N. Each user considers that the values of t_p and t_c may have been modified by the actions of previous users. The user U_i now has the possibility to choose where to allocate her shiftable load. For each possibility, her payoff function is calculated, taking into account the total production and consumption through the grid at each time unit. After this, U_i allocates her shiftable load in the time unit which maximizes her payoff function q_i. Performing this step on all the users of the considered set is called an *iteration*.

3. For each user, a check is performed in order to see if any user has changed the allocation of her shiftable load compared to the previous iteration. If none has, the simulation ends. If at least one user has changed it, the game goes back to step 2 with the current allocations.

4. After a certain maximum number of iterations, or if the configuration of the shiftable loads is identical to the one reached in a previous iteration, the game ends.

If the game ends at step 3, the configuration of shiftable loads has reached a Nash equilibrium, since the most profitable strategy for each user is to leave the load in the allocation where it already was. If the game ends at step 4 for finding a configuration already known in the past, the game will never reach a Nash equilibrium, since it will go on a loop. It is not necessary to establish a maximum number of iterations, as the number of configurations is finite and therefore the game will eventually end in one of the two ways that have been just described. However, if computational time gets too high, it may be convenient to choose a limit for those.

These are the cases that have been simulated in order to see whether the algorithm just described converges or not.

- Payoff functions are determined by the original NRG-X-Change selling and buying functions. (**Original**)

- Payoff functions are determined by the selling and buying functions defined in Chapter 4. (**Improved**)

- Payoff functions are determined by the selling function in Eq. 5.11 and the buying function in Eq. 5.13. (**Log/Sq)**

- Payoff functions are determined by the selling function in Eq. 5.12 and the buying function in Eq. 5.14. (**SqRs**)

Since the functions' parameters are arbitrary, they have been chosen so that for $t_p = t_c$ all the selling functions have the same value, and all the buying functions have the same value , which roughly correspond to the average selling/buying (respectively) tariffs already existing for the Cardiff grid. In Table 5.1 we report the number of times the game converges to a NE depending on the percentage of energy producers through the grid. Table 5.2 describes the average number of iterations needed to obtain convergence with each couple of functions **(Iterations)**, and the average increase of self-consumed energy through the grid after performing the game **(SelfCons)**.

From these results, these considerations follow.

- The functions labeled as **Original** and **Improved** do not guarantee that the algorithm converges to a NE. On the other hand, as expected from the theoretical results, the new proposed functions always converge to a Nash equilibrium.

%	Original	Improved	Log/Sq	SqRs
10%	20	203	245	245
30%	114	232	245	245
50%	242	245	245	245

Table 5.1: Number of cases in which Nash equilibrium is reached depending on the used selling/buying functions. The number of simulations carried out is 245 for each case and couple of functions; the size of the grid varies, ranging from 10 to 70 users. The leftmost column indicates the percentage of grid users who produce energy. The functions are, in order from left to right: the original NRG-X-Change functions, the functions proposed in Chapter 4, the logarithm selling and square buying functions, and the square root selling and negative square root buying functions.

Functions	Iterations	SelfCons
Original	4.7255	15.7248
Improved	4.8554	20.2876
Log/Sq	4.0370	23.2854
SqRs	3.8872	23.3021

Table 5.2: Results of the simulations in terms of convergence and self-consumption. The first column describes the functions, the second column indicates the average number of iterations for reaching a NE, and the third column reports how much, on average, the amount of self-consumed energy increases with our optimization.

- The **Improved** functions perform better than the **Original** functions in terms of convergence, for every considered grid size and percentage of prosumers. They also provide better results in terms of increase of self-consumption, although the average number of iterations required for convergence is slightly higher.
- The new proposed functions (**SqRs** and **Log/Sq**) outperform the old functions (**Original** and **Improved**) on all the considered aspects. They always guarantee convergence to a NE, need less iterations to converge compared to the old functions, and also improve self-consumption through the grid.
- Among the new proposed functions, the *SqRs* functions seems to perform better than the *Log/Sq* functions in terms of convergence speed. Also, the amount of self-consumed energy is, on average, slightly higher.

Finally, we report in graphical form the consequences of the game to the daily net consumption through the grid, which achieves an effect of peak-shaving/valley-filling. We took as example the case with 4 prosumers and 36 consumers, although all cases present the same behavior. In Figure 5.1 the difference between production and consumption of energy is shown at each time unit, before and after optimization.

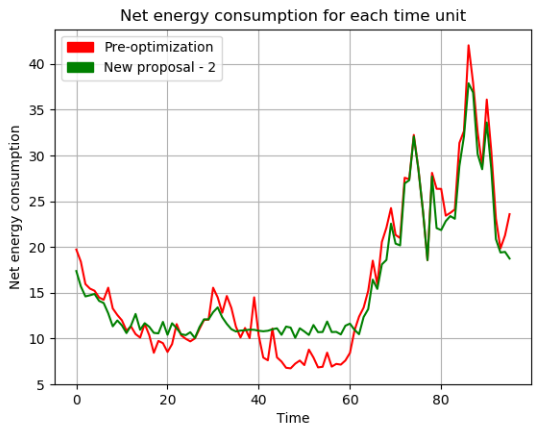


Figure 5.1: Graph of net consumption of energy before and after the optimization. The x axis indicates the time units through the day, while on the y axis is indicated the difference between total consumption and production through the grid. We have depicted the values before the optimization (in red) and after the optimization (in green). It has to be remarked that the shiftable loads were randomly allocated through the day in the initial configuration.

Chapter 6

Multiagent Systems for cost and congestion management: the Woerden scenario.

6.1 Context

Local energy communities (LECs) are considered a promising way to integrate distributed energy resources and to engage end-users in sustainable energy practices. In LECs, participants can locally buy and sell their energy. When production inside the community is not sufficient to meet local demand, the electricity shortage is covered by import from the main grid. On the other hand, the surplus energy can be exported to the grid. By acting together, community members have a stronger negotiation power when interacting with other energy market participants. The trade inside the community is encouraged by the difference between local and retail prices. Trading locally is beneficial because it allows funds remain within local economy. This also reduces losses that occur when the energy is transmitted over long distances.

The recent interest in LECs contributed to a growing number of industrial projects and research publications [72]. Some focus on the design of a peer-to-peer (P2P) market with necessary functions where peers can buy and sell their energy. Thus, [55] proposed a virtual currency to regulate energy exchange between peers, via the NRG-X-Change mechanism, which we have seen in the previous chapters. The payment functions for those supplying and consuming energy were developed to encourage the energy balance. The approach is dependent on function settings [2] and does not account for a strategic behavior of peers. The latter is often addressed using a game theory. [73] proposed price-based schemes and a game-theoretical framework to coordinate flexible demand. Though, distributed generation and energy storage systems were not taken into consideration. [74] suggested an auction-based market mechanism, where each household provides bids or offers of their demand or generation. These offers and bids are collected to allocate energy and determine prices. In the auction based market, accurate estimates of energy and prices are needed to get better tariffs. This can be disadvantageous to inexperienced participants. A major advantage of P2P schemes is that they are able to preserve privacy as there is no central supervisory entity.

A more structured market design enables a cost sharing mechanism where community members pay in the form of a share of a single electricity bill of the overall community. This design assumes the coordination of actions towards the common goal. Thus, [75] proposed a two stage aggregated control to realize a P2P energy sharing where an energy sharing coordinator controls flexible devices. An interior-point method was used to minimize the energy costs of the community. A centralized in nature is its major disadvantage that is also common to many existing approaches based on energy sharing. The centralization limits scalability and rises concerns about privacy. Multiagent systems (MAS) are promising alternative to decentralize computations. [76] surveyed MAS applications for microgrid control. Energy trading in the community using MAS was recently addressed by [77]. Although this approach presented an agent-based model of the community, a limited number of flexible devices and only the flow of electricity were considered.

This study presents a multiagent system developed to manage a community of households located in the central Netherlands. Our approach is close to cost sharing ones. We describe the developed agent-based model of the community energy network. This model accounts for the flows of electrical and heat energy. The model conveniently decomposes the underlying optimization problem so that each agent is assigned with a specific task that can be solved efficiently. In particular, we present an agent-based decomposition for addressing the model of battery with losses. This contrasts with existing approaches. For example, [78] addressed the problem directly introducing additional parameter for optimization. Furthermore, the presented MAS approach is not limited to the considered community and can be adapted to other real-world scenarios.

6.2 Case study

The present study has been carried out in the scope of a European project whose aim is to unlock a demand response potential in the distribution grid. The addressed case study involves a community of households located in the central Netherlands. Figure 6.1 depicts the architecture of community and aggregator platform. This figure presents a logical architecture rather than the physical one. This view is applicable to manage: (i) households located in a close neighborhood sharing a common point of connection with the main grid, as shown in Figure 6.1 and (ii) households that are geographically distant and have no physical common point coupling.

The considered community consists of 16 households and a district battery with the capacity of 220 kWh. Each house has a heat pump of 2 kW combined with a hot water buffer of 200L. The heat pump is used for heating both domestic hot water and spatial heating. Each house has solar photovoltaics (PV) capable of producing up to 7kW. Additionally, there is one in-home battery with the capacity of 7.8 kWh.

The given community is managed by an energy service providing company know as

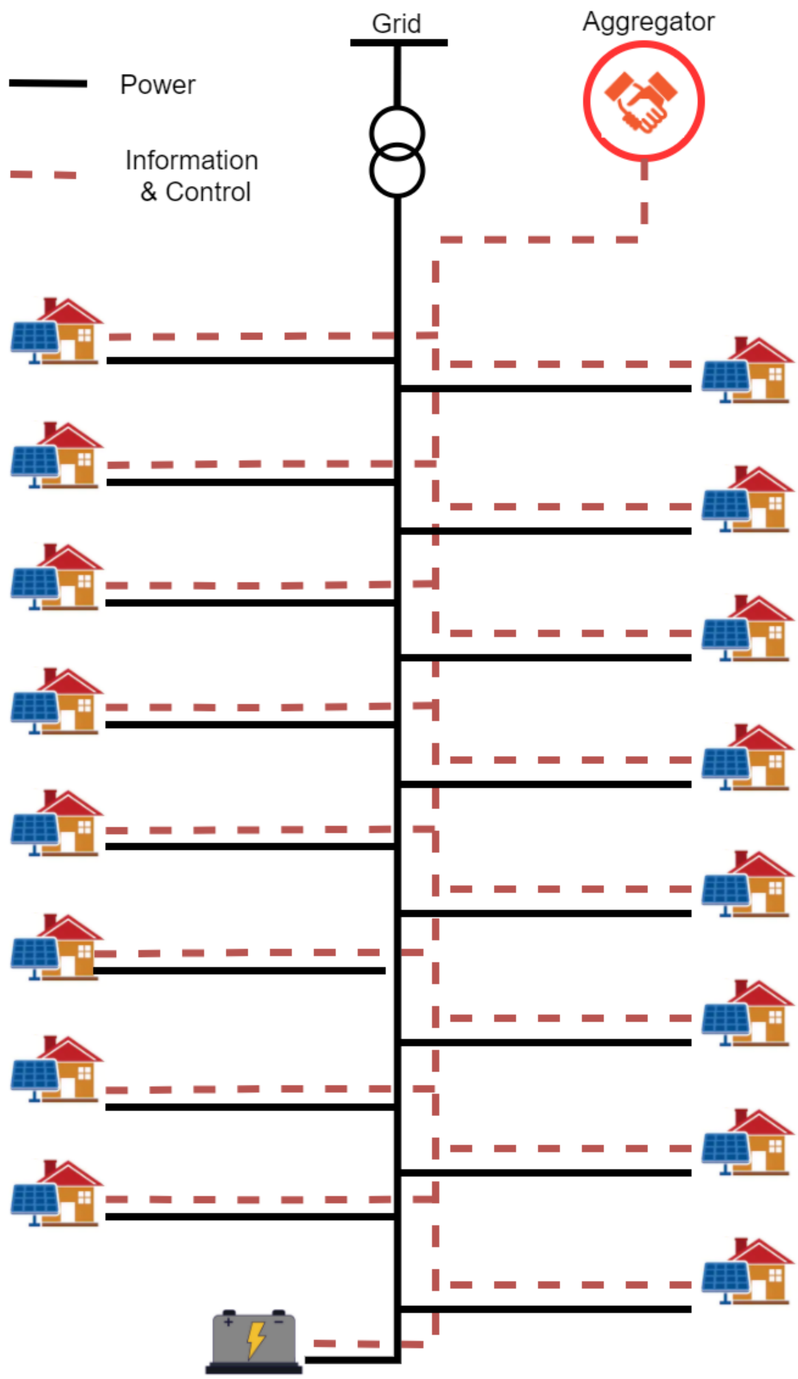


Figure 6.1: Architecture of the energy community.

aggregator. The aggregator manages flexible assets in the community aiming to maximize the value of flexibility. The aggregator platform facilitates the interaction with external parties and provides an infrastructure for executing MAS. It relies on information and communications technology and consists of different communication devices, software applications and protocols. Each house is equipped with a local energy gateway (LEG) that is connected with flexible devices and is able to communicate with the backend run on a cloud. Thus, LEGs are used to control flexible devices and to communicate sensory data. The Advanced Message Queuing Protocol is used for communication.

6.3 Multiagent system

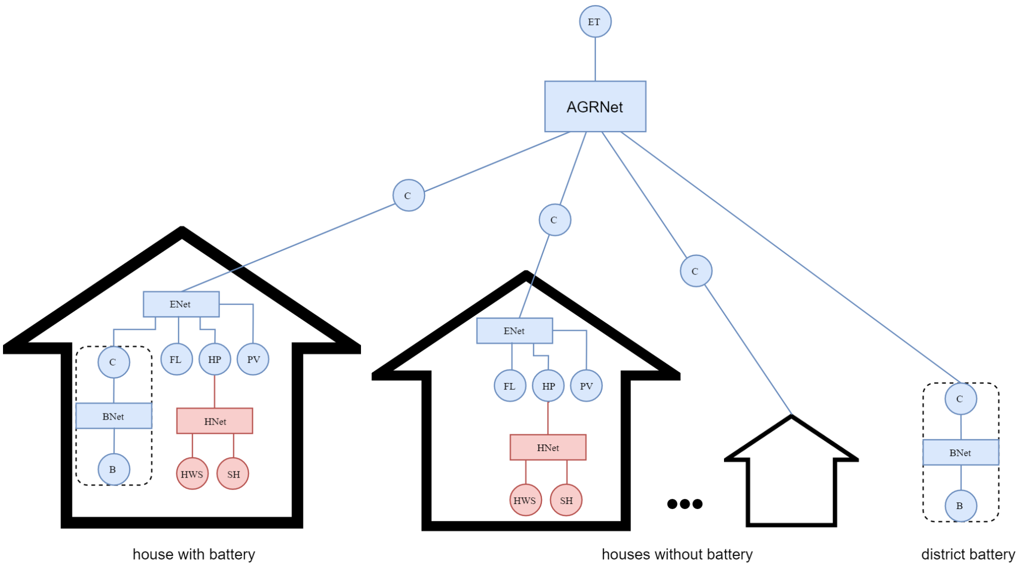


Figure 6.2: Multiagent model of the community energy network.

The community energy network is modeled as a multiagent system where agents are nodes and the environment is everything outside the agent. The state of the environment is represented by sensory data including the charge levels of batteries, indoor and outdoor temperatures, temperature limits for comfort, the prices of energy and possible limits for energy flows. The actions of agents are the amount of energy pulled from or injected to the grid.

6.3.1 Multiagent model

Figure 6.2 shows the multiagent model of the community energy network. The energy network is represented by a bipartite graph having two set of nodes. This suggests two types of agents, shown by circles and rectangles. Circles depict device agents (D). The device can refer to an abstract or physical device. Rectangles show net agents (N). The

Table 6.1: Agent types and their roles

Notation	Description	Role
AGRNet	Aggregator Net	AGRNet represents a local energy market. It aggregates different assets and ensures the energy balance at the community level.
ENet	Electrical Net	ENet represents the electrical network in the house energy system. It ensures the balance of electrical energy.
HNet	Heating Net	HNet represents the heating network in the house energy system. It ensures the balance of heat energy.
BNet	Battery Net	BNet ensures the energy balance in the decomposed model of battery.
ET	External Tie	ET represents a connection to an external source of power. This can be viewed as a connection between the community and the main grid.
C	Connector	C connects two nets modelling the transmission of energy. The transmission can be associated with losses. There are no losses between AGRNet and ENet. The transmission loss from ENet to BNet models a charging efficiency. The transmission loss from BNet to ENet models a discharging efficiency.
B	Battery	B represents an electrical storage that can take in or deliver energy.
PV	Photovoltaic	PV represents solar panels that generate electricity from absorbing sunlight.
FL	Fixed Load	FL represents an inflexible energy consumption that must be satisfied.
HP	Heat Pump	HP transforms electricity to heat energy with some conversion coefficient.
SH	Space Heating	SH represents indoor air temperature that must be kept within limits for comfort.
HWS	Hot Water Storage	HWS represents the tank with hot water and the consumption of domestic hot water.

net represents a virtual zone where the energy exchange between the devices takes place. Edges in the graph indicate interaction between agents. Table 6.1 lists different types of agents and explains their roles.

The model represents a structure of agent interactions. It extends a multiagent model of a building by aggregating different buildings and district battery using AGRNet. This net can be viewed as a local energy market where district actors negotiate their energy exchange. AGRNet is also connected to an external source of power that represents a connection point between the community and the main grid. Inside a building, there are electrical (shown in blue) and heating (shown in red) networks that account for the flows of electricity and heat. Net agents ensure the balance of corresponding energies. Connector agents that connect houses to the AGRNet represent prosumers in the local energy market. They communicate schedules for houses while keeping energy flows inside houses hidden at the community level. The battery with losses is represented by connector, net and battery agents. The agent that connects to the battery net models losses associated with charging and discharging. The battery agent models linear constraint associated with its capacity.

This problem decomposition has been performed to obtain an adequate representation of the energy network such that individual subproblems are easy to solve and yield meaningful solutions.

6.3.2 Optimization problem

MAS addresses the problem of minimizing the energy bill at the connection point while satisfying energy balance constraints, physical constrains of devices and comfort preferences. Using a splitting technique for the variables associated with net and device agents the following multiagent optimization problem is formulated

$$\begin{aligned} \underset{x_i \in \Omega_i, z_i \in \Theta_i}{\text{minimize}} \quad & \sum_{i \in D} f_i(x_i) + \sum_{i \in N} g_i(z_i) \\ \text{subject to} \quad & x_i = z_i, \ \ \forall i \in N \end{aligned} \tag{6.1}$$

where f_i is a real valued objective function associated with the i-th device and defined in the feasible region Ω_i, g_i is a real valued objective function associated with the i-th net and defined in the feasible region Θ_i, x_i and z_i are the decision variables associated with the i-th devices and net, respectively. The constraints account for the fact that respective device and net agents should agree upon the values of shared variables.

Finding the minimizer to an equality constrained optimization problem is equivalent to identifying the saddle point of the associated Lagrangian function. This gives the following augmented Lagrangian function

$$L(x,z,\lambda) = \sum_{i \in D} f_i(x_i) + \sum_{i \in N} g_i(z_i) + \frac{\rho}{2} \sum_{i \in N} \|x_i - z_i + u_i\|_2^2 \tag{6.2}$$

where $u_i = \lambda_i/\rho$ is the scaled dual variable (for Lagrange multipliers λ_i^T) , x_i and z_i are primal variables.

6.3.3 Agent coordination

Agents solve their local problems in a coordination with their neighbors. The coordination mechanism is based on the alternating direction method of multipliers (ADMM). ADMM iteratively minimizes the augmented Lagrangian function (6.2) with respect to primal and dual variables. Each iteration involves the following steps.

Step 1. Device agents compute in parallel their optimal variables by solving

$$x_{i\in D}^{(k+1)} = \arg\min_{x_i\in\Omega_i} f_i(x_i) + \frac{\rho}{2}\|x_i - (z_i^k - u_i^k)\|_2^2 \tag{6.3}$$

The corresponding values are communicated to neighboring nets.
Step 2. Net agents compute in parallel their optimal variables by solving

$$z_{i\in N}^{(k+1)} = \arg\min_{z_i\in\Theta_i} g_i(z_i) + \frac{\rho}{2}\|z_i - (x_i^{(k+1)} + u_i^k)\|_2^2 \tag{6.4}$$

Step 3. Net agents in parallel update their dual variables.

$$u_{i\in N}^{(k+1)} = u_i^k + (x_i^{(k+1)} - z_i^{(k+1)}) \tag{6.5}$$

The corresponding primal and dual variables are sent to neighboring devices.

The above steps are repeated until convergence criteria are met. The convergence criteria are defined locally for primal and dual residuals as

$$\begin{aligned} r^{\text{primal}} &< \varepsilon^{\text{primal}} \\ r^{\text{dual}} &< \varepsilon^{\text{dual}} \end{aligned} \tag{6.6}$$

where $\varepsilon^{\text{primal}}, \varepsilon^{\text{dual}}$ are small positive numbers representing primal and dual tolerances, respectively. The primal and dual residuals are computed as

$$\begin{aligned} r^{\text{primal}} &= \|x_i^k - z_i^k\|_2 \\ r^{\text{dual}} &= \|\rho(z_i^k - z_i^{k-1})\|_2 \end{aligned} \tag{6.7}$$

If the device and net functions $f(x)$ and $g(z)$ are convex, the constraint residual under ADMM is guaranteed to converge to zero and the objective value to the minimum of the dual problem, see [79].

6.3.4 Agent models

Nets

The net agents (AGRNet, ENet, HNet and BNet) ensure the energy balance treating the constraints

$$\sum_{i\in D} z_i(\tau) = 0, \quad \tau = 1,\ldots,H.$$

The problem (6.4) is solved by projecting the variables received from neighboring devices onto the feasible region as in [80].

Devices

The C agent connects two nets. In each time step τ, there two possible cases:
(i) the energy flows from net 1 to net 2, then

$$0 \le x_1(\tau) \le x_1^{\max}(\tau) \quad \text{and} \quad \eta_1 \cdot x_1(\tau) = -x_2(\tau)$$

(ii) the energy flows from net 2 to net 1, then

$$0 \le x_2(\tau) \le x_2^{\max}(\tau) \quad \text{and} \quad -x_1(\tau) = \eta_2 \cdot x_2(\tau)$$

where $\eta_1, \eta_2 \in (0,1]$ are transmission efficiencies. The problem (6.3) is solved by finding critical points for two cases and selecting the one with a lower value of (6.3) in each time step τ. HP represents a particular case when the energy flows only from ENet to HNet.

The ET agent is associated with the following objective function

$$f(x) = \sum_{\tau=1}^{H} p^{\text{buying}}(\tau) \cdot x^{+}(\tau) + \sum_{\tau=1}^{H} p^{\text{selling}}(\tau) \cdot x^{-}(\tau)$$

where $p^{\text{buying}}(\tau)$ and $p^{\text{selling}}(\tau)$ are respectively the price of imported and exported energy, $x^{+}(\tau)$ is the energy with the positive sign (imported), and $x^{-}(\tau)$ is the energy with the negative sign (exported). The set of constraints restrict the energy coming from and into the grid.

$$x^{\min} \le x(\tau) \le x^{\max}, \quad \tau = 1, \ldots, H.$$

where $x^{\min}(\tau)$ and $x^{\max}(\tau)$ are respectively the minimum and the maximum energy flow allowed. The problem (6.3) is solved by finding critical points for positive and negative cases and selecting the one with a lower value of (6.3). Constraints are treated by projection.

The FL and PV agents aim to satisfy respectively the forecast consumption and production

$$x(\tau) = \hat{x}(\tau), \quad \tau = 1, \ldots, H$$

where $\hat{x}$ are forecast values. The solution to (6.3) is trivial ($x^k = \hat{x}$).

The B agent has a set of constraints aiming to keep its state of charge as well as charging and discharging rates within the allowed range.

$$Q^{\min} \le Q(\tau) \le Q^{\max}, \quad \tau = 1, \ldots, H$$

$$x^{\min} \le x(\tau) \le x^{\max}, \quad \tau = 1, \ldots, H$$

where $Q^{\min}$ and $Q^{\max}$ are the minimum and maximum allowed charge of the battery, $x^{\min}$ and $x^{\max}$ are limits for discharging and charging rates. The battery's charge evolves as [80]

$$Q(\tau) = Q^{\text{init}} + \sum_{\tau=1}^{H} x(\tau)$$

where Q^{init} is the initial charge of the battery.

The SH agent has a set of constraints to ensure a room temperature is within the comfort temperature limits.

$$T^{\min} \leq T(\tau) \leq T^{\max}, \quad \tau = 1, \dots, H$$

where $T^{\min}$ and $T^{\max}$ are the temperature limits. The room temperature evolves as [80]

$$T(\tau) = T(\tau-1) + \frac{\mu}{c} \cdot \Big(T^{\text{amb}}(\tau) - T(\tau-1)\Big) + \\ + \frac{\eta}{c} \cdot x(\tau), \quad \tau = 1, \dots, H$$

where $T(0)$ is the initial temperature, T^{amb} is the outdoor temperature, μ is the conduction coefficient, η is the heating efficiency and c is the heat capacity of indoor air.

The HWS agent has a set of constraints to ensure the temperature of water inside the tank is within the allowed range.

$$T^{\min} \leq T(\tau) \leq T^{\max}, \; \tau = 1, \dots, H$$

with $T^{\min}$ and $T^{\max}$ are the temperature limits. The water temperature evolves as [81]

$$T(\tau) = T(\tau-1) + \frac{V_{\text{cold}}(\tau)}{V_{\text{total}}} \cdot \Big(T_{\text{cold}} - T(\tau-1)\Big) + \\ + \frac{1}{V_{\text{total}} \cdot c} \cdot x(\tau), \quad \tau = 1, \dots, H$$

where $T(0)$ is the initial temperature, V_{cold} is the volume of water with temperature T_{cold} entering the tank to replace the consumed hot water, V_{total} is the tank volume, and c is the specific heat of water. The consumption of hot water is forecast.

The B, SH and HWS agents solve the problem (6.3) using Dykstra's projection method with a starting point $(z^k - u^k)$.

6.3.5 Agent negotiation

ADMM iteratively processes primal and dual variables of the augmented Lagrangian function. These variables have a meaningful interpretation within the energy network. The primal variables represent the energy and the dual variables define its price in different nodes across the network.

Agent interactions, governed by ADMM, model a negotiation process where device agents negotiate their energy exchange in exchange zones represented by net agents. Messages sent from nets to devices represent requests. These messages contain the amount of energy requested, z_i^k. The sign is used to distinguish between production and consumption. From a recipient perspective, a positive value indicates that the energy flows towards the recipient, a negative value indicates the flow towards the sender. It also includes the

Table 2: Different market situations

u_i^k	z_i^k	Market situation
Positive	Positive	Overconsumption/Underproduction: the market is proposing the prosumer to consume the energy z_i paying the price u_i.
Positive	Negative	Overconsumption/Underproduction: the market is proposing the prosumer to produce the energy z_i and get paid the price u_i.
Negative	Positive	Underconsumption/Overproduction: the market is proposing the prosumer to consume the energy z_i and get paid the price u_i.
Negative	Negative	Underconsumption/Overproduction: The market is proposing the prosumer to produce the energy z_i paying the price u_i.

price, u_i^k, which not only indicates the current market price, but also is used to distinguish between a situation of overconsumption (in which the price will be positive) and a situation of underconsumption (in which the price will be negative).

Table 2 lists different market situations in exchange zones of energy network based on the sign of the price and the requested power. The first two cases (i.e. positive prices) refer to a market with underproduction (i.e. overconsumption) situation. The last two cases (i.e. negative prices) refer to a market with overproduction (i.e. underconsumption). In all cases, the suggested payment is given by $u_i^k \cdot z_i^k$.

The device agents send messages with offers as response to received requests. This establishes the precedence relation in agent communication meaning that offer messages are only sent after request messages have been received and processed. An offer message contains the amount of energy, x_i^k, offered by the device in each time interval.

The negotiation continues until the consensus is reached between the agents. The device agents agree on their energy profiles so that their variables are equal to those requested by the nets. This situation means the supply is equated to the demand in all the exchange zones and the corresponding markets are cleared.

6.4 Results and Discussion

This section presents the results obtained by the developed MAS. The tests were performed using the data for the winter season. Winter days are characterized by the need to use heat pump for both heating domestic hot water and maintaining room temperature within comfort limits. This represents the most challenging optimization scenario as during other seasons heat pump is only used for reheating domestic hot water.

6.4.1 Optimizing community cost

Community members participate in community energy management having a common interest of increasing social welfare through collective use of local resources. This implies collaboration in terms of energy use to reduce the cost at the connection point when there is no external flexibility requests.

Day ahead optimization

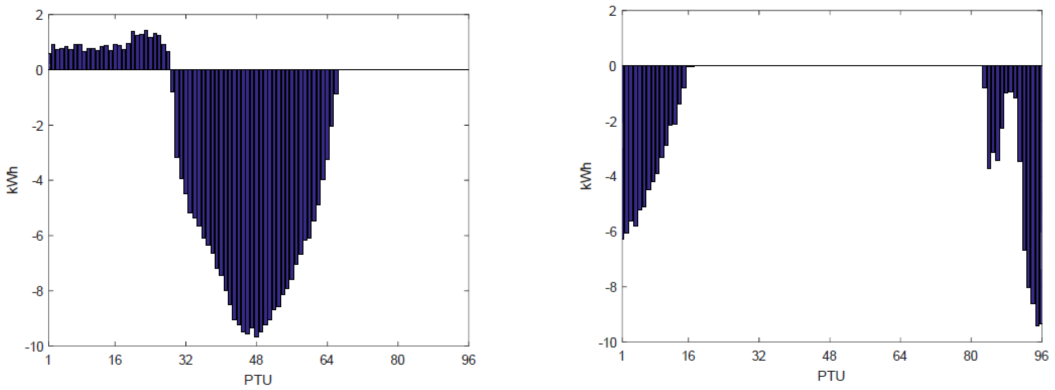


Figure 6.3: Energy profiles of the community when optimization starts at 00:00 (left) and 12:00 (right)

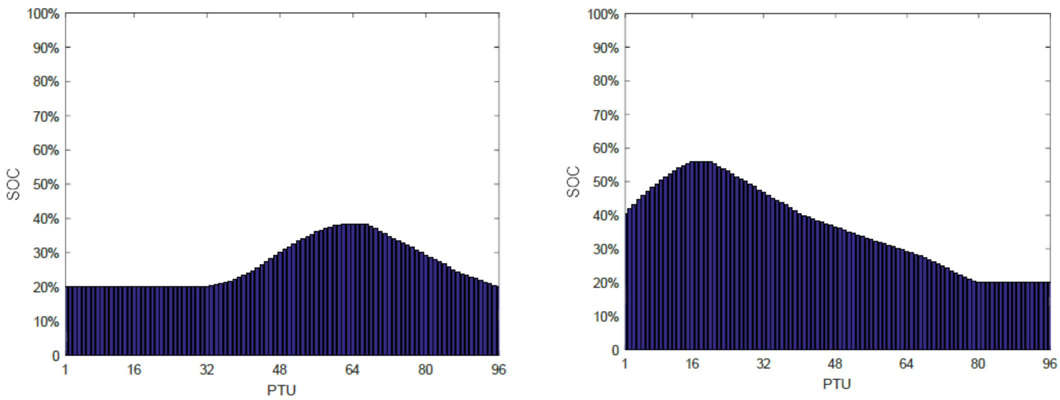


Figure 6.4: SOC of the district battery when optimization starts at 00:00 (left) and 12:00 (right).

Multiagent optimization suggests optimal energy usage for the next 24 hours discretized into 96 program time units (PTUs) corresponding to 15 minute time intervals. When running a day ahead optimization, the system attempts to schedule energy consumption for flexible loads to time intervals when renewable energy from solar panels is available. Starting optimization at different time slots during the day changes the perspective as the time horizon moves forward embracing new time slots from the next day. Figure 6.3 provides insights about the effect of executing optimization at different PTUs during the same day. The figure shows the energy profiles of the community at the connection point with the main grid. Positive values refer to the amount of energy taken from the grid (imported energy). Negative values indicate the energy fed into the grid (exported energy). The plot on the left shows results when optimization was executed at midnight. The plot on the right refers to starting time at midday which embraces the data for the following day.

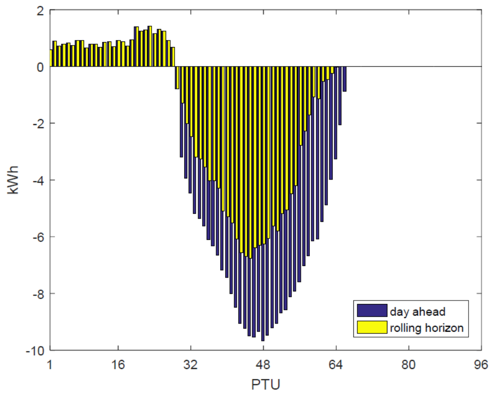
The battery allows storing the excess of renewable energy and later delivering this energy by discharging when local demand is high. This is why the community does not import energy during evening hours. The local demand is met by the energy released from the battery. MAS schedules the district battery charging and discharging so that it takes locally produced energy only in the amount needed to meet local demand until the end of the time horizon. Figure 6.4 provides much insight about this process. The plots show state of charge (SOC) with optimization starting at different PTUs. It was considered that in the beginning the battery had the lowest allowed SOC of 20%. During the day the SOC is increased by injecting the energy from solar PVs. Later this energy is released to supply local demand. No extra energy is left or released for export. Without additional constraints for optimization, this is the most effective strategy and is as expected. Because the use of battery is always associated with some costs due to charging and discharging efficiencies, the battery should only be used when that is really necessary.

Rolling horizon optimization

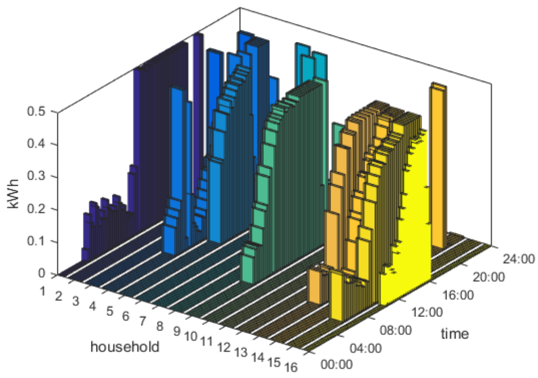
The method of applying optimization at every program time unit is known as rolling horizon optimization. Rolling horizon can mitigate uncertainties in the models, such as forecast errors, and account for the fact that the real time horizon is not limited to a single day.

Figure 6.5 summarizes the results for rolling horizon optimization. The effect of rolling horizon optimization can be understood from Figure 6.5a. For comparison, this figure shows the community energy profiles obtained by day ahead and rolling horizon optimizations. The former provides the results from the single optimization run at the beginning of the day. The latter involves the results of 96 optimizations during the day, with the data for the first PTU being used for control. The difference can be readily understood. Rolling horizon optimization results in smaller energy exports during times of intense electricity generation from PV panels. This is explained by the fact that rolling horizon accounts for time slots in the next days with a need to meet local demand.

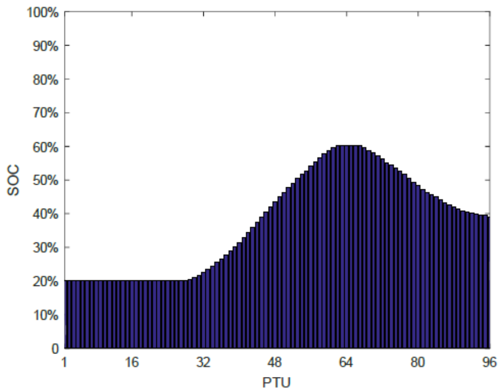
The day for which results are presented in Figure 6.5a is characterized by a massive production from solar PVs panels. The optimized energy profiles suggest peaks in times of high PV generation. This is because no constraints on possible power flows were imposed. In such scenario, the system solely focuses on minimizing the energy bill of the community. Although the results indicate that rolling horizon optimization can reduce peaks to some extent, it is not the goal of the community at this step. The issue of dealing with possible grid overload is addressed in the next section as part of congestion management scenarios.



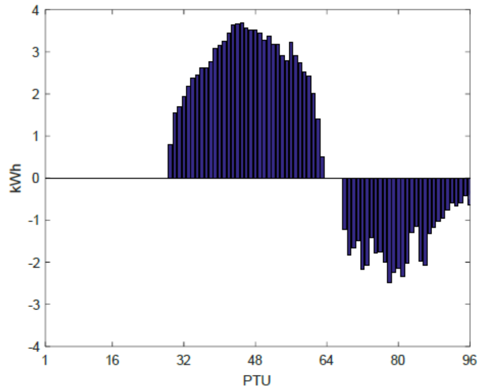
(a) Energy profile of the community



(b) Energy profiles of heat pumps



(c) SOC of the district battery



(d) Charging and discharging of the district battery

Figure 6.5: Results for rolling horizon optimization.

The optimal behavior of the community is characterized by shifting consumption of flexible devices to time intervals with electricity available from solar panels. This is because this electricity is cheaper than the one taken from the main grid. Thus, both in-home and district devices are expected to use energy in such times during the day.

Heat pumps are flexible devices at the home level. They are used to provide heat for both domestic hot water and space heating. As expected, MAS suggests using heat pumps in times with available renewable energy. This is shown in Figure 6.5b that depicts a three-dimensional bar chart with the consumption profiles of each heat pump in the community. It can be seen that most of consumption occurs in times of high PV generation. Consumption in other time intervals is dictated by the need to satisfy temperature limits, especially for domestic hot water. The consumption of domestic hot water is irregular with respect to time and amount, especially in the evening.

At the district level, there is a battery storage. When multiagent optimization runs in a rolling horizon mode, in the end of the day, the battery SOS is not at its lower state

Table 6.2: Results for community optimization.

	Optimization	
	In-home	Community
Energy import (kWh)	16072.94	14384.53
Energy export (kWh)	3694.83	1814.42
Self-consumption (%)	59	80

because it holds some energy to account for the following time horizon. This is shown in Figure 6.5c that summarizes 96 optimization runs for one day. Figure 6.5d graphically illustrates the control actions for the battery during the day in terms of the amount of energy charged and discharged in each PTU.

In-home vs community optimization

Community management should ensure its members benefit from collective energy usage. To provide insights about the advantages of community-based approach we compared the results of in-home and community optimizations. The former optimized community households individually and combined their energy profiles. The latter performed the community optimization. The results indicate the performance of two approaches varies depending on the amount of renewable energy that is locally produced. Community-based approach is better when there is excess of renewable energy. On the other hand, both approaches perform similarly when it is low.

Figure 6.6 illustrates the amount of imported energy for the two optimization approaches. It can be seen that for all days community optimization yields less or equal energy import as those of in-home optimization. Similar results are only observed when PV generation is low. The analysis of the results indicates that an increase in renewable energy generation leads to a decrease in energy imported from the grid.

Table 6.2 further summarizes the obtained results. During the period under consideration, the total renewable energy production was 9047.91 kWh. Similarly to the above daily data, the total energy import and export of the community was reduced. It is important to note that besides economic benefits this also reduces the stress on the main grid. Instead of exporting the produced renewable energy it is used to meet local demand.

Self-consumption is an important performance indicator. It is defined as the ratio between the energy consumed and the total energy produced by the community. As shown in Table 6.2, the community based optimization approach increases self-consumption. These results provide important insights about advantages of collective energy management and the estimates of potential savings.

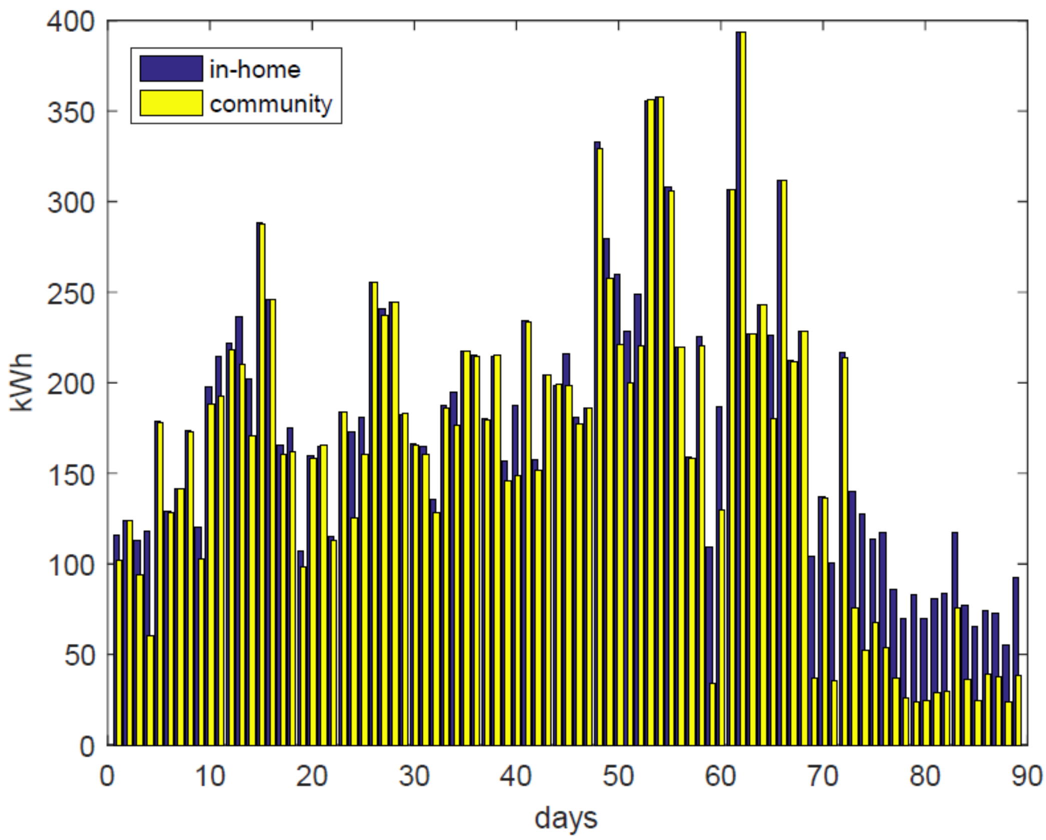


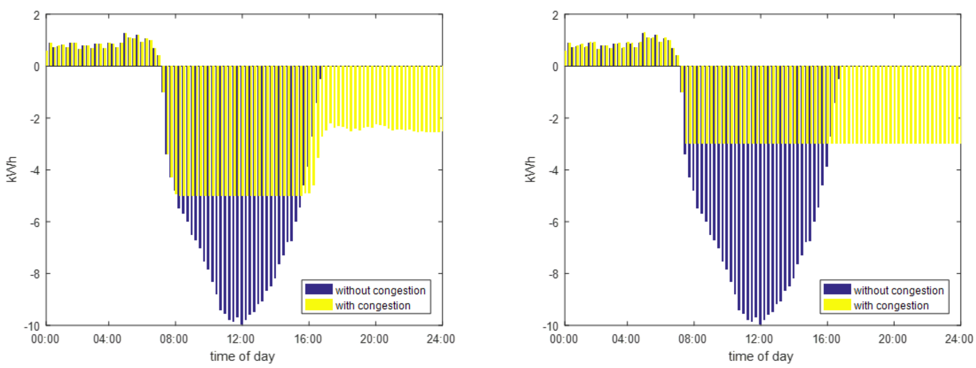
Figure 6.6: Comparing the energy import of the community when using in-home optimization individually and community optimization.

6.4.2 Congestion management

Power volatility caused by renewables and irregular consumption poses significant challenges for the distribution grid. This volatility has the potential to lead to network congestion, decreasing reliability. Congestion management refers to avoiding the thermal overload of system components by reducing loads. Congestion occurs if the capacity of the grid is insufficient to accommodate the requested power flows.

This section presents and discusses the results of addressing two different congestion management scenarios. In these scenarios, first the optimal community energy profiles were found by community optimization without considering grid capacity constraints. Next, it was considered that a grid safety analysis performed by a distribution system operator (DSO) required to limit the injected/consumed energy. The limits were introduced into the MAS in the form of constraints. The resulting constrained optimization problem was solved by multiagent optimization to reschedule the energy use in the community.

Overproduction



(a) maximum export 5kWh (b) maximum export 3kWh

Figure 6.7: Community energy profiles under congestion management scenarios due to overproduction.

The first considered congestion management scenario refers to the situation in the grid with an excess of injected renewable energy. This can readily occur in sunny weather when local demand is low. Under such circumstances, an increase in voltage can happen and grid components can be damaged if no proper actions are taken.

Figure 6.7 shows a day ahead scenario for the community addressing the oversupply of renewable energy. The energy profiles resulting from unconstrained community optimization are shown in blue. The results obtained when addressing flexibility requests from the DSO are shown in yellow. Two different requests limiting the energy injection are considered. These correspond to the maximum energy injection of 5 kWh and 3 kWh per PTU. It can be seen that both grid capacity limits are respected by the MAS. The

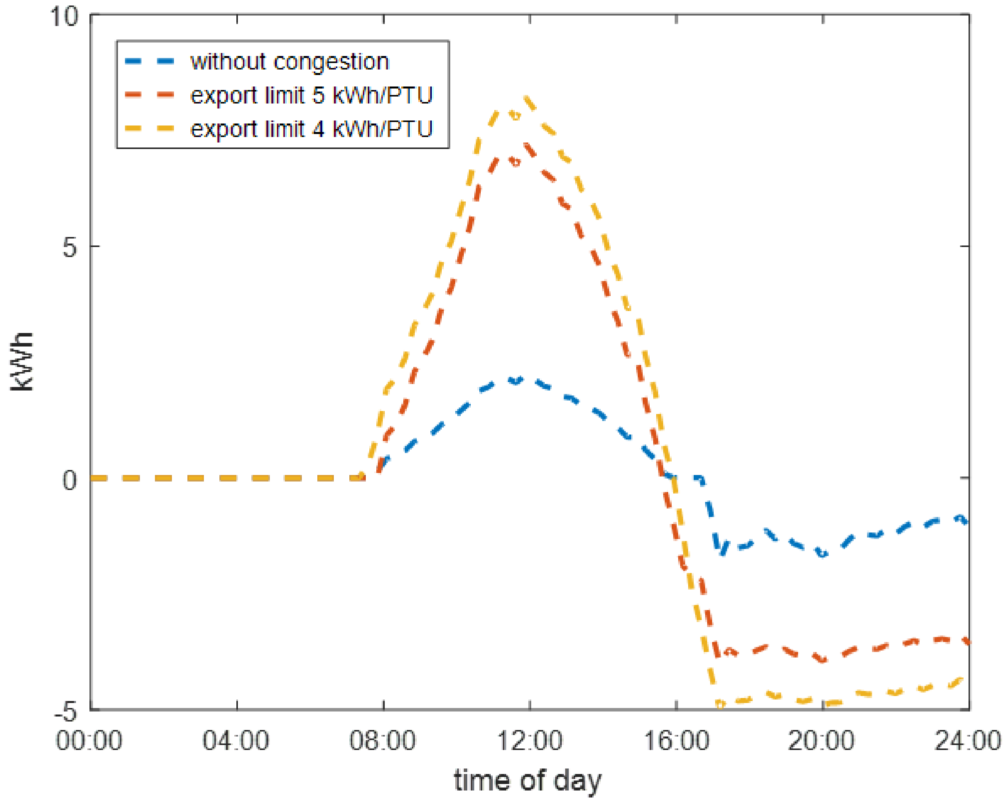
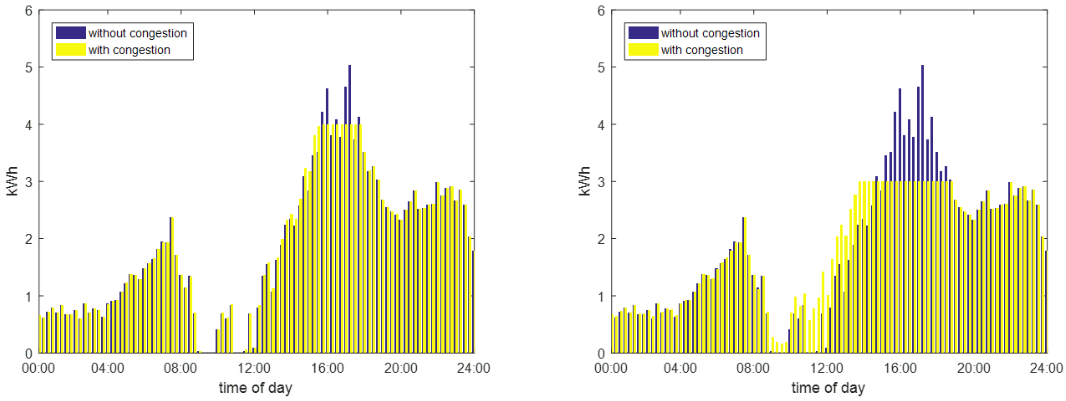


Figure 6.8: Charging and discharging of community storage with different constraints limiting community export.

amount of renewable energy injected into the main grid does not exceed the limits requested. This is achieved by shifting the export from peak sunny periods during the day to evening and night hours. This reduces the stress on the distribution grid and provides energy in times of high demand, such as evening hours.

The availability of the storage capacity in the community is critical to exhibiting the above behavior. The battery is able to take the excess of renewable energy in peak times and later release it. Since the community is interested in minimizing the cost and maximizing the profit, the losses in the battery should be minimized.

Figure 6.8 shows the behavior of community storage in terms of energy charging and discharging in this scenario. This figure illustrates that the more limited grid capacity, the more energy is taken in peak times of PV production and correspondingly the more energy released later. This results are as expected and confirm the ability of MAS to adapt its energy export according to grid conditions.



(a) maximum import 4kWh

(b) maximum import 3kWh

Figure 6.9: Community energy profiles under congestion management scenarios due to overconsumption.

Overconsumption

The second considered congestion management scenario involves the overload of grid components due to high demand. This situation is common in times with limited local production and high consumption.

Figure 6.9 depicts the community energy profiles addressing these congestion management scenarios. The plots show the results for requests limiting peak consumption to 4 kWh and 3 kWh per PTU. For comparison, the plots also show results for community optimization without congestion constraints. Similarly to previously considered scenario, these results suggest that MAS is able to schedule the energy use for the community under considered congestion management scenarios.

The plots show that high demand occurs in evening hours resulting in a peak consumption around 16:00. The requests from the DSO ask for lowering this demand. To address this the community needs to anticipate peak demand and to use the energy available in off-peak times. The more power reduction is requested the more energy is consumed during off-peak periods. This energy is taken by the community battery storage when the local demand is low. Later, during peak times, the stored energy is released. Thus, the community does not overload the grid pulling energy in critical periods and meet its demand by using the locally stored energy.

Figure 6.10 shows charging and discharging of community storage without and with congestion management. Without congestion, no energy is injected. This is because during the considered day there is no excess of renewable energy. All the energy generated by rooftop panels is consumed locally. However, when MAS takes into consideration DSO requests, the battery storage is utilized to charge in off-peak time intervals and discharge later to meet peak demand.

The reduction in peak demand can also be addressed by shifting consumption of flexible loads. Heat pumps are flexible loads in the considered community. Figure 6.11 shows

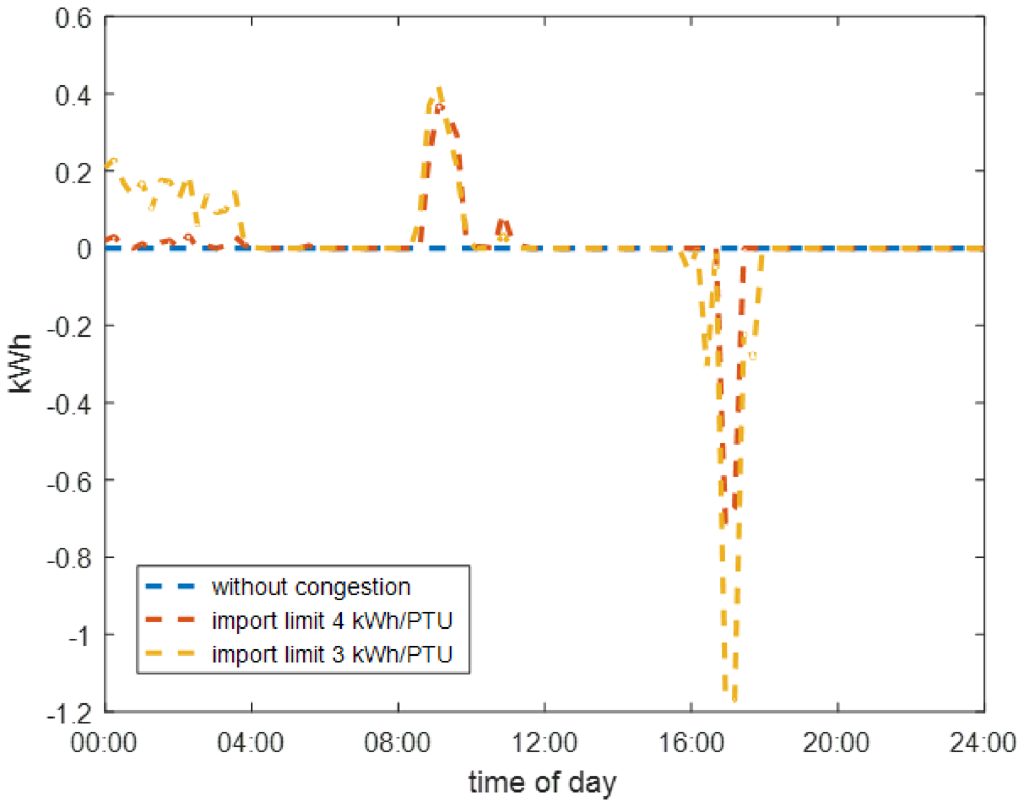


Figure 6.10: Charging and discharging of community storage with different constraints limiting community import.

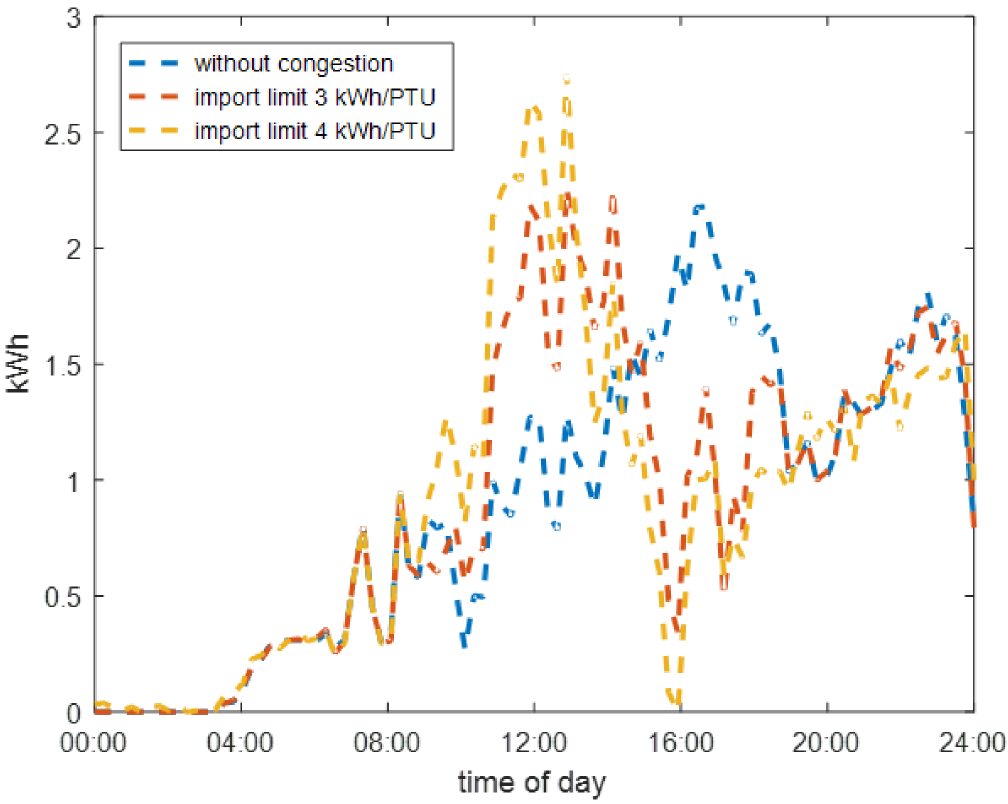


Figure 6.11: Heat pumps consumption in the community with different constraints limiting community import.

the cumulative consumption of heat pumps in the community. A shift in the consumption from peak to off-peak periods can be observed. This shift takes place alongside with the use of battery storage. The energy consumed by heat pumps can come from the battery and from the exported energy as long as constraints are respected.

Thus, the obtained results show that flexibility requests from the DSO are met through efficient use of flexible loads, which combines charging and discharging of battery storage as well as shifting heat pump consumption to anticipate a peak load.

Chapter 7

Conclusions

With the increasing diffusion of smart grids, one of the most important new concepts that have been introduced is flexibility. In this thesis we have shown various examples of its relevance in smart grid contexts and how its usage can be exploited in order to build mathematical models, which enable us to solve some of the most important problems in this new energy grid context.

First, our work in demand response highlights the use of flexibility in smart grids by showing how it can be employed to minimize energy cost and discomfort for the prosumers, as well as to meet the needs of external actors such as BRPs, DSOs and aggregators. The optimization is performed at various levels: first, within the context of a single prosumer/house, second, based on interactions between prosumers interested in buying/selling energy, and, finally, based on external entities interested in buying prosumers' flexibility. The combined effort of those levels' optimizations can bring significant economic advantages to the energy market players, although in this thesis we focus more on the prosumers.

In particular, the cost and discomfort of prosumers are reduced on average by up to 29% compared to the results obtained by the naive algorithm (which corresponds to our baseline), At the same time, the peak load can be reduced on average by over 11% (with respect to the baseline) as a result of the third level of optimization. Furthermore, the exploitation of parallel computation in level 1 may reduce computational time by over 92%, and in addition to this, using it in level 2 may further reduce computational time by over 51%, ensuring that the time needed to perform the complete optimization is less than half a minute. The employment of parallel computing and dedicated hardware (GPUs) allows getting rid of various approximations and exploiting the optimal solutions, which is very important especially in real scenarios.

In our work on pricing systems, flexibility is what realizes the purpose of the pricing functions to shift consumption from one time to another, and therefore determines the incentive system's effectiveness. Our work has exposed some important issues in the state-of-the-art NRG-X-Change mechanism, and addressed them by proposing new functions to determine reward for energy production, and cost for energy consumption. These functions efficiently solve the issues presented, by actively discouraging curtailment in energy production for prosumers, taking into account situations where a congestion may occur, and encouraging prosumers self-consumption (instead of selling all their produced energy and buy the energy they need for consumption from another source).

Local energy communities and peer-to-peer energy trading are expected to be important components of future energy systems. Our work in the game theory section investigated different mechanisms for regulating energy exchanges between agents in local energy communities. The agents are assumed to act in their own self-interest when optimizing energy usage. The market mechanisms are designed to incentivize production and self-consumption of locally produced renewable energy while matching supply and demand. In this case, other than having the usage of the previous chapter, flexibility is even more at the core of the mechanism, as flexible loads define the strategies of each player in the game we have described.

The proposed design exhibits advantageous theoretical properties when compared with some existing market mechanisms. The results obtained by simulations using real-world data reveal that incentives offered by the proposed mechanisms in the form of buying and selling prices lead more sustainable behaviours of energy communities. Exploiting their flexible loads, consumers schedule their consumption in periods with an intensive injection of renewable energy whereas producers are given a high reward that help discouraging them from curtailment, which can appear under other approaches.

In the last chapter, this work presented a multiagent system for a real-world scenario of community energy management. An appropriate multiagent model was developed considering electrical and heating networks. The agent models were presented indicating solution methods for their subproblems.

The obtained results offered important insights and provided the evidence for the validity of multiagent system approach. Single run of multiagent optimization results in optimal control actions for all controllable devices in the community. These are in the form of consumed and/or injected energy as well as temperature set points for thermal loads. The system demonstrated the ability to offer solutions satisfying all the constraints whenever feasible solutions exist. These constraints involve allowed state of charge of battery, temperature limits in each time unit and energy balance in each node of the network. The suggested consumption profiles are meaningful. The energy consumption for flexible loads is scheduled in times when renewable energy is available. The export of energy is scheduled so that local demand is met first and only excess of local supply is fed into the main grid. This is how flexibility has been employed by this novel approach in order to overcome the issues that have been treated.

Future work on these problems can go in several directions. Regarding the demand response solution proposed in this thesis, what can improve it significantly would be implementing: (i) forecasting algorithms for energy consumption and generation; (ii) forecasting algorithms for real time tariffs; and (iii) different flexibility settings. In particular, adding those features to the created framework would make it an even more complete demand response tool.

Regarding the improved incentive mechanism, a promising line of future work would be to explore different options for the penalty functions that control congestion in our scheme. Moreover, it would be interesting to perform further experiments by simulation of other grid environments. Regarding the game theory approach, we intend to investigate how the proposed mechanisms behave when diverse sources of flexibility are available including heating systems and storage units. Finally, a promising future direction to enhance the multi-agent system approach would be treating the issue of a fair distribution of costs and profits between community members.

Bibliography

[1] Fabio Lilliu, Andrea Loi, Diego Reforgiato Recupero, Mario Sisinni, and Meritxell Vinyals. An uncertainty-aware optimization approach for flexible loads of smart grid prosumers: A use case on the cardiff energy grid. *Sustainable Energy, Grids and Networks*, 20:100272, 12 2019.

[2] Fabio Lilliu, Meritxell Vinyals, Roman Denysiuk, and Diego Reforgiato Recupero. A novel payment scheme for trading renewable energy in smart grid. In *Proceedings of the Tenth ACM International Conference on Future Energy Systems*, e-Energy '19, pages 111–115, 2019.

[3] Roman Denysiuk, Fabio Lilliu, Diego Reforgiato Recupero, and Meritxell Vinyals. Peer-to-peer energy trading for smart energy communities. In Ana Paula Rocha, Luc Steels, and H. Jaap van den Herik, editors, *Proceedings of the 12th International Conference on Agents and Artificial Intelligence, ICAART 2020, Volume 1, Valletta, Malta, February 22-24, 2020*, pages 40–49. SCITEPRESS, 2020.

[4] Roman Denysiuk, Fabio Lilliu, Meritxell Vinyals, and Diego Reforgiato Recupero. Multiagent system for community energy management. In Ana Paula Rocha, Luc Steels, and H. Jaap van den Herik, editors, *Proceedings of the 12th International Conference on Agents and Artificial Intelligence, ICAART 2020, Volume 1, Valletta, Malta, February 22-24, 2020*, pages 28–39. SCITEPRESS, 2020.

[5] Michele Moretti, Sylvestre Njakou Djomo, Hossein Azadi, Kristof May, Kristof De Vos, Steven Van Passel, and Nele Witters. A systematic review of environmental and economic impacts of smart grids. *Renewable and Sustainable Energy Reviews*, 68(Part 2):888 – 898, 2017.

[6] Usef: An introduction to the universal smart energy framework, `https://www.usef.energy/app/uploads/2016/12/An-introduction-to-the-Universal-Smart-Energy-Framework-2.pdf`, 2015.

[7] Mario Sisinni, Ivan Grimaldi, Christope Ponchel, Sébastien Breton, David Pires, Vincenzo Giordano, Thomas Messervey, and Juan Espeche. D6.1 - detailed use cases scenarios. *MAS^{2}TERING public reports*, May 2016.

[8] Francisco Javier Ruiz-Rodriguez, Jesus Hernandez, and Francisco Jurado. Probabilistic load flow for radial distribution networks with photovoltaic generators. *IET Renewable Power Generation*, 6(2), 2012.

[9] Alessandro Agnetis, Gabriella Dellino, Gianluca De Pascale, Giacomo Innocenti, Marco Pranzo, and Antonio Vicino. Optimization models for consumer flexibility aggregation in smart grids: The address approach. *2011 IEEE First International Workshop on Smart Grid Modeling and Simulation (SGMS)*, pages 96–101, Oct 2011.

[10] Sarah O'Connell and Stefano Riverso. Flexibility analysis for smart grid demand response. *CoRR*, abs/1704.01308, 2017.

[11] Mario Sisinni, Hassan Sleiman, Meritxell Vinyals, Youssef Oualmakran, Yann Pankow, Ivan Grimaldi, Camillo Genesi, Juan Espeche, Michael Dibley, Baris Yuce, Hisain Elshaafi, and Steve McElveen. D2.2 - mas^{2}tering platform design document. *MAS^{2}TERING public reports*, Febrary 2016.

[12] Usef: The framework explained, `http://www.globalsmartgridfederation.org/wp-content/uploads/2016/10/USEF_TheFrameworkExplained-18nov15.pdf`, 2015.

[13] Patrick Lynch, Thomas Messervey, Juan Espeche, Zia Lennard, Mario Sisinni, Richard Hickey, Josephine Holohan, Jamie Power, and Derek Kelly. D1.5 - business strategies and collaboration opportunities. *MAS^{2}TERING public reports*, August 2016.

[14] Ahsan Raza Khan, Anzar Mahmood, Awais Safdar, Zafar A. Khan, and Naveed Ahmed Khan. Load forecasting, dynamic pricing and dsm in smart grid: A review. *Renewable and Sustainable Energy Reviews*, 54(Supplement C):1311 – 1322, 2016.

[15] Akin Tascikaraoglu, Ali Boynuegri, and Mehmet Uzunoglu. A demand side management strategy based on forecasting of residential renewable sources: A smart home system in turkey. *Energy and Buildings*, 80(Supplement C):309 – 320, 2014.

[16] Stephen P. Holland and Erin T. Mansur. Is real-time pricing green? the environmental impacts of electricity demand variance. *The Review of Economics and Statistics*, 90(3):550–561, 2008.

[17] Peter Palensky and Dietmar Dietrich. Demand side management: Demand response, intelligent energy systems, and smart loads. *IEEE Trans. Industrial Informatics*, 7(3):381–388, 2011.

[18] Antimo Barbato and Antonio Capone. Optimization models and methods for demand-side management of residential users: A survey. *Energies*, 7:5787–5824, 2014.

[19] Amir Hamed Mohsenian Rad and Alberto Leon-Garcia. Optimal residential load control with price prediction in real-time electricity pricing environments. *IEEE Trans. Smart Grid*, 1(2):120–133, 2010.

[20] Mohammad Rastegar, Mahmud Fotuhi-Firuzabad, and Farrokh Aminifar. Load commitment in a smart home. *Applied Energy*, 96:45 – 54, 2012. Smart Grids.

[21] Christopher O. Adika and Lingfeng Wang. Autonomous appliance scheduling for household energy management. *IEEE Trans. Smart Grid*, 5(2):673–682, 2014.

[22] Xiaodao Chen, Tongquan Wei, and Shiyan Hu. Uncertainty-aware household appliance scheduling considering dynamic electricity pricing in smart home. *IEEE Trans. Smart Grid*, 4(2):932–941, 2013.

[23] Edgar Galván-López, Tom Curran, James McDermott, and Paula Carroll. Design of an autonomous intelligent demand-side management system using stochastic optimisation evolutionary algorithms. *Neurocomputing*, 170(Supplement C):270 – 285, 2015. Advances on Biological Rhythmic Pattern Generation: Experiments, Algorithms and Applications Selected Papers from the 2013 International Conference on Intelligence Science and Big Data Engineering (IScIDE 2013) Computational Energy Management in Smart Grids.

[24] Amin Fakhrazari, Hamid Vakilzadian, and F. Fred Choobineh. Optimal energy scheduling for a smart entity. *IEEE Trans. Smart Grid*, 5(6):2919–2928, 2014.

[25] Edgar Galván-López, Colin Harris, Leonardo Trujillo, Katya Rodriguez-Vazquez, Siobhán Clarke, and Vinny Cahill. Autonomous demand-side management system based on monte carlo tree search. *2014 IEEE International Energy Conference (ENERGYCON)*, pages 1263–1270, May 2014.

[26] Carolina M. Affonso and Luiz C.P. da Silva. Potential benefits of implementing load management to improve power system security. *International Journal of Electrical Power & Energy Systems*, 32(6):704 – 710, 2010.

[27] Pengwei Du and Ning Lu. Appliance commitment for household load scheduling. *IEEE Trans. Smart Grid*, 2(2):411–419, 2011.

[28] Manuel Castillo-Cagigal, Alvaro Gutiérrez, Felix Monasterio-Huelin, Estefanía Caamaño-Martín, D. Masa, and Javier Jiménez-Leube. A semi-distributed electric demand-side management system with pv generation for self-consumption enhancement. *Energy Conversion and Management*, 52(7):2659 – 2666, 2011.

[29] Saskia Lavrijssen and Arturo Carrillo Parra. Radical prosumer innovations in the electricity sector and the impact on prosumer regulation. *Sustainability*, 2017.

[30] Yuan Luo, Satoko Itaya, Shin Nakamura, and Peter Davis. Autonomous cooperative energy trading between prosumers for microgrid systems. pages 693–696, 2014.

[31] Cynthujah Vivekananthan, Yateendra Mishra, Gerard Ledwich, and Fangxing Li. Demand response for residential appliances via customer reward scheme. *IEEE Trans. Smart Grid*, 5(2):809–820, 2014.

[32] Ditiro Setlhaolo, Xiaohua Xia, and Jiangfeng Zhang. Optimal scheduling of household appliances for demand response. *Electric Power Systems Research*, 116(Supplement C):24 – 28, 2014.

[33] Zhuang Zhao, Won-Cheol Lee, Yoan Shin, and Kyung-Bin Song. An optimal power scheduling method for demand response in home energy management system. *IEEE Trans. Smart Grid*, 4(3):1391–1400, 2013.

[34] Hamid Manoochehri and Alireza Fereidunian. A multimarket approach to peak-shaving in smart grid using time-of-use prices. pages 707–712, 2016.

[35] Pedram Samadi, Amir-Hamed Mohsenian-Rad, Robert Schober, Vincent W. S. Wong, and Juri Jatskevich. Optimal real-time pricing algorithm based on utility maximization for smart grid. In *2010 First IEEE International Conference on Smart Grid Communications*, pages 415–420, Oct 2010.

[36] Hamed Shakouri G. and Aliyeh Kazemi. Multi-objective cost-load optimization for demand side management of a residential area in smart grids. *Sustainable Cities and Society*, 32(Supplement C):171 – 180, 2017.

[37] William Burke and David Auslander. Residential electricity auction with uniform pricing and cost constraints. In *41st North American Power Symposium*, pages 1–6, Oct 2009.

[38] Dongsik Jang, Jiyong Eom, Min Jae Park, and Jae Jeung Rho. Variability of electricity load patterns and its effect on demand response: A critical peak pricing experiment on korean commercial and industrial customers. *Energy Policy*, 88:11 – 26, 2016.

[39] Payam Teimourzadeh Baboli, Mehdi Eghbal, M. Parsa Moghaddam, and Habib Allah Aalami. Customer behavior based demand response model. *2012 IEEE Power and Energy Society General Meeting*, pages 1–7, July 2012.

[40] UK national grid. Future energy scenarios - uk gas and electricity transmission, July 2015.

[41] Future energy scenarios - uk gas ans energy transmission, `http://fes.nationalgrid.com/media/1292/2016-fes.pdf`, 2016.

[42] Hassan Sleiman, Meritxell Vinyals, Sandra Garcia Rodriguez, Loïs Vanhee, Michael Dibley, Shaun Howell, Jean-Laurent Hippolyte, Yacine Rezgui, and Julien Ardeois. D3.1 - multi-agent systems holonic platform generic components. *MAS^{2}TERING public reports*, May 2016.

[43] Nadeem Javaid, Fahim Ahmed, Ibrar Ullah, Samia Abid, Wadood Abdul, Atif Alamri, and Ahmad S. Almogren. Towards cost and comfort based hybrid optimization for residential load scheduling in a smart grid. *Energies*, 10:1546, 2017.

[44] Duy Long Ha, Stephane Ploix, Eric Zamai, and Mireille Jacomino. A home automation system to improve household energy control. *IFAC Proceedings Volumes*, 9(3):15–20, 2006.

[45] Liyan Jia and Lang Tong. Optimal pricing for residential demand response: A stochastic optimization approach. pages 1879–1884, 2012.

[46] Enrico Carpaneto and Gianfranco Chicco. Probability distributions of the aggregated residential load. *Probabilistic Methods Applied to Power Systems, 2006. PMAPS 2006. International Conference on*, 2006.

[47] Engy A. Mohamed and Yasser G. Hegazy. A novel probablistic strategy for modeling photovoltaic based distributed generators. *World Academy of Science, Engineering and Technology International Journal of Energy and Power Engineering*, 9(8), 2015.

[48] Marc Wenninger, Jochen Schmidt, and Toni Goeller. Appliance usage prediction for the smart home with an application to energy demand side management - and why accuracy is not a good performance metric for this problem. pages 143–150, 2017.

[49] Felipe Feijoo, Walter Silva, and Tapas K. Das. A computationally efficient electricity price forecasting model for real time energy markets. *Energy Conversion and Management*, 113:27 – 35, 2016.

[50] Mohammad Reza Ranjbar and Sajjad Kouhi. Sources' response for supplying energy of a residential load in the form of on-grid hybrid systems. *International Journal of Electrical Power & Energy Systems*, 64(Supplement C):635 – 645, 2015.

[51] Hafiz Majid Hussain, Nadeem Javaid, Sohail Iqbal, Qadeer Ul Hasan, Khursheed Aurangzeb, and Musaed Alhussein. An efficient demand side management system with a new optimized home energy management controller in smart grid. *Energies*, 11(1), 2018.

[52] Mihail Mihaylov, Roxana Rădulescu, Iván Razo-Zapata, Sergio Jurado, Leticia Arco, Narcís Avellana, and Ann Nowé. Comparing stakeholder incentives across state-of-the-art renewable support mechanisms. *Renewable Energy*, 131:689–699, 2019.

[53] The European Commission. European commission guidance for the design of renewables support schemes, in: Delivering the internal market in electricity and making the most of public intervention. Technical report, The European Commission, 2013. `https://ec.europa.eu/energy/sites/ener/files/com_2013_public_intervention_swd04_en.pdf`.

[54] Dejan Ilic, Per Goncalves da Silva, Stamatis Karnouskos, and Martin Griesemer. An energy market for trading electricity in smart grid neighbourhoods. In *6th IEEE International Conference on Digital Ecosystems and Technologies, DEST 2012, Campione d'Italia, Italy, June 18-20, 2012*, pages 1–6, 2012.

[55] Mihail Mihaylov, Sergio Jurado, Narcís Avellana, Kristof Van Moffaert, Ildefons Magrans de Abril, and Ann Nowé. Nrgcoin: Virtual currency for trading of renewable energy in smart grids. In *European Energy Market (EEM), 2014 11th International Conference on the*, pages 1–6. IEEE, 2014.

[56] J. K. Kok, C. J. Warmer, and I. G. Kamphuis. Powermatcher: multiagent control in the electricity infrastructure. In *4rd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2005), July 25-29, 2005, Utrecht, The Netherlands - Special Track for Industrial Applications*, pages 75–82, 2005.

[57] Nicola Capodieci, Giuliano Andrea Pagani, Giacomo Cabri, and Marco Aiello. Smart meter aware domestic energy trading agents. In *First International E-Energy Market Challenge Workshop (co-located with ICAC'11)*. ACM Press, 2011. Relation: https://www.rug.nl/fmns-research/bernoulli/index Rights: University of Groningen, Johann Bernoulli Institute for Mathematics and Computer Science.

[58] Mihail Mihaylov, Sergio Jurado, Narcís Avellana, Iván S. Razo-Zapata, Kristof Van Moffaert, Leticia Arco, Maite Bezunartea, Isel Grau, Adrian Cañadas, and Ann Nowé. SCANERGY: a scalable and modular system for energy trading between prosumers. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2015, Istanbul, Turkey, May 4-8, 2015*, pages 1917–1918, 2015.

[59] Georgios Methenitis, Michael Kaisers, and Han La Poutré. Incentivizing intelligent customer behavior in smart-grids: A risk-sharing tariff & optimal strategies. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 380–386, 2016.

[60] Meritxell Vinyals, Valentin Robu, Alex Rogers, and Nicholas R. Jennings. Prediction-of-use games: a cooperative game theory approach to sustainable energy tariffs. In *International conference on Autonomous Agents and Multi-Agent Systems, AAMAS '14, Paris, France, May 5-9, 2014*, pages 829–836, 2014.

[61] Valentin Robu, Meritxell Vinyals, Alex Rogers, and Nicholas R. Jennings. Efficient buyer groups with prediction-of-use electricity tariffs. *IEEE Trans. Smart Grid*, 9(5):4468–4479, 2018.

[62] Hongyao Ma, Valentin Robu, Na Li, and David C. Parkes. Incentivizing reliability in demand-side response. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 352–358, 2016.

[63] Hongyao Ma, David C. Parkes, and Valentin Robu. Generalizing demand response through reward bidding. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2017, São Paulo, Brazil, May 8-12, 2017*, pages 60–68, 2017.

[64] Reshef Meir, Hongyao Ma, and Valentin Robu. Contract design for energy demand response. *CoRR*, abs/1705.07300, 2017.

[65] Smart Grid Task Force. Regulatory recommendations for the deployment of flexibility. *EU SGTF-EG3 Report*, 2015.

[66] Walid Saad, Zhu Han, H. Vincent Poor, and Tamer Basar. Game-theoretic methods for the smart grid: An overview of microgrid systems, demand-side management, and smart grid communications. *IEEE Signal Process. Mag.*, 29(5):86–105, 2012.

[67] Amir Hamed Mohsenian Rad, Vincent W. S. Wong, Juri Jatskevich, Robert Schober, and Alberto Leon-Garcia. Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid. *IEEE Trans. Smart Grid*, 1(3):320–331, 2010.

[68] Hung Khanh Nguyen, Ju Bin Song, and Zhu Han. Demand side management to reduce peak-to-average ratio using game theory in smart grid. In *2012 Proceedings IEEE INFOCOM Workshops, Orlando, FL, USA, March 25-30, 2012*, pages 91–96, 2012.

[69] Hazem M. Soliman and Alberto Leon-Garcia. Game-theoretic demand-side management with storage devices for the future smart grid. *IEEE Trans. Smart Grid*, 5(3):1475–1485, 2014.

[70] A. Loni and F. Parand. A survey of game theory approach in smart grid with emphasis on cooperative games. In *2017 IEEE International Conference on Smart Grid and Smart Cities (ICSGSC)*, pages 237–242, July 2017.

[71] J. B. Rosen. Existence and uniqueness of equilibrium points for concave n-person games. *Econometrica*, 33(3):520–534, 1965.

[72] Tiago Sousa, Tiago Soares, Pierre Pinson, Fabio Moret, Thomas Baroche, and Etienne Sorin. Peer-to-peer and community-based markets: A comprehensive review. *Renewable and Sustainable Energy Reviews*, 104:367 – 378, 2019.

[73] A. De Paola, D. Angeli, and G. Strbac. Price-based schemes for distributed coordination of flexible demand in the electricity market. *IEEE Transactions on Smart Grid*, 8:3104–3116, 2017.

[74] P. Shamsi, H. Xie, A. Longe, and J. Joo. Economic dispatch for an agent-based community microgrid. *IEEE Trans Smart Grid*, 7(5):2317–24, 2016.

[75] C. Long, J. Wu, Y. Zhou, and N. Jenkins. Peer-to-peer energy sharing through a two-stage aggregated battery control in a community microgrid. *Applied Energy*, 226:261–276, 2018.

[76] A. Kantamneni, L. E. Brown, G. Parker, and W. W. Weaver. Survey of multi-agent systems for microgrid control. *Engineering Applications of Artificial Intelligence*, 45:192–203, 2015.

[77] Meritxell Vinyals, Maxime Velay, and Mario Sisinni. A multi-agent system for energy trading between prosumers. In *14th International Conference on Distributed Computing and Artificial Intelligence*, pages 79–86, 2018.

[78] P. Yang and A. Nehorai. Joint optimization of hybrid energy storage and generation capacity with renewable energy. *IEEE Transactions on Smart Grid*, 5(4):1566–1574, 2014.

[79] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, January 2011.

[80] Matt Kraning, Eric Chu, Javad Lavaei, and Stephen Boyd. Dynamic network energy management via proximal message passing. *Found. Trends Optim.*, 1(2):73–126, January 2014.

[81] Mohammad Tasdighi, Hassan Ghasemi, and Ashkan Rahimi-Kian. Residential microgrid scheduling based on smart meters data and temperature dependent thermal load modeling. *IEEE Transactions on Smart Grid*, 5:349–357, 01 2014.

Funding

La borsa di dottorato è stata cofinanziata con risorse del
Programma Operativo Nazionale Ricerca e Innovazione 2014-2020 (CCI 2014IT16M2OP005),
Fondo Sociale Europeo, Azione I.1 “Dottorati Innovativi con caratterizzazione Industriale”

UNIONE EUROPEA
Fondo Sociale Europeo

Ministero dell'Università
e della Ricerca

PON
RICERCA
E INNOVAZIONE
2014 - 2020