Extending PowerMatcher to Reduce Peak Demand Created by Electric Vehicles

by

Iain Meeke

Dissertation

Presented to the

University of Dublin, Trinity College

in fulfilment

of the requirements

for the Degree of

Master in Computer Science

University of Dublin, Trinity College

Supervisor: Dr. Donal O'Mahony

May 2017

Declaration of Authorship

I, Iain Meeke, declare that the following dissertation, except where otherwise stated, is entirely my own work; that it has not previously been submitted as an exercise for a degree, either in Trinity College Dublin, or in any other University; and that the library may lend or copy it or any part thereof on request.

Iain Meeke

May 17, 2017

Acknowledgements

I would like to thank Donal O'Mahony for his supervision, guidance and continuous enthusiasm. Donal constantly helped me pick things back up when they got difficult.

Thank you to my family for their support and providing a warm home, this year and in all my years of education.

Finally thank you to my classmates, without whom I would not have made it through the past five years.

Iain Meeke
University of Dublin, Trinity College

May 2017

Summary

The structure of the electricity grid has remained the same since it was first developed in the late 18th Century. This structure consists of central generation, a large transmission network, and a radial distribution network. The majority of power stations at the centre of this structure burn fossil fuels to generate electricity. Burning fossil fuels releases carbon dioxide into the atmosphere, causing climate change. The world has become more aware of the effects of climate change, causing a rise in popularity in generating electricity using renewable sources such as wind and solar. Renewable electricity generation tends to be on a smaller scale than traditional generation, and therefore it is distributed throughout the electricity grid. Renewable energy sources are also intermittent. Indeterminacy and distribution require huge efforts to correctly coordinate the electricity grid and balance the supply and demand of the grid.

The worry of climate change has also increased the popularity of electric vehicles. Electric vehicles do provide a benefit in that they can use energy more efficiently than a combustion engine vehicle. However electric vehicles will create a huge strain on the electricity grid as there will be a large increase in overall demand. Without coordination this big demand would require an increase in supply to the grid. However this may not be possible without replacing the ageing grid network to handle an increase in capacity.

Demand response is a solution to these problems. Demand response is a method of reducing demand to match the available supply of electricity. There are multiple techniques for demand response, however the majority of them require some form of central manual control. The new distributed grid will require automatic coordination. Transactive energy is a control mechanism that will allow real time coordination of the electricity grid. It uses value as an operational parameter to allow distributed agents to control devices connected to the electricity grid.

PowerMatcher is open-source software that provides a transactive energy technique for smart grid coordination. It is a multi-agent system comprising of three main agent types. Each energy source or energy consuming device is represented by a device agent, while the other two agents perform coordination of the system. These agents communicate using bid and price messages based on transactive energy techniques. Device agents communicate immediate energy needs for the current time slot in real time. While this means PowerMatcher can balance supply and demand in real time, it does not provide intelligent coordination of large future loads.

Electric vehicles are an example of large loads that are predictable to a certain degree. With current functioning capabilities, PowerMatcher will still cause large demand peaks if there is too much demand in a system. This danger could be reduced by adding two new message types for prediction and allocation of large loads, ensuring the system remains in balance.

This dissertation investigates PowerMatcher as a solution to problems outlined above. It provides simulations and device agents for devices that make up the modern neighbourhood. These simulations made it apparent that a prediction mechanism could be added to PowerMatcher to extend it's capabilities.

Extending PowerMatcher to Reduce Peak Demand Created by Electric Vehicles

Iain Meeke, Master in Computer Science
University of Dublin, Trinity College, 2017

Supervisor: Dr. Donal O'Mahony

The current electricity grid has remained the same since it was first developed. It provides reliable electricity by burning fossil fuels at large central power stations. Renewable energy sources are becoming more popular and tend to be distributed, with small generation spread throughout the electricity grid. These distributed renewable sources add complexity to the grid as they are intermittent and difficult to coordinate. PowerMatcher is a tool that provides real-time distributed coordination of the electricity grid. PowerMatcher is a multi-agent system in which agents communicate using bid and price messages based on transactive energy techniques. This research investigates PowerMatcher's behaviour in a modern neighbourhood scenario. In particular, the ability to balance supply and demand of neighbourhood and the peak to average demand is assessed. Simulation of electric vehicles, solar panels, batteries, and wind turbines were created to test PowerMatcher's ability. A potential improvement for PowerMatcher is identified during these simulations that would allow PowerMatcher to balance future loads as well as acting in real-time. The design of the extension for this improvement is described in this research.

Contents

List of Figures

Chapter 1

Introduction

1.1 Motivation and Research Objectives

The electricity grid is rapidly changing. Currently, the grid relies on centralised fossil fuel generation but as people become increasingly aware of the negative effects of these energy sources, the demand for alternatives rises. These alternatives typically come in the form of distributed renewable energy. As the price of photovoltaic (PV) panels falls and the efficiency increases, more people are installing residential panels [8]. Although an increase in solar and other renewable sources can reduce the demand on the electricity grid, they are intermittent and add a layer of complexity. The electric grid in Ireland supplies energy at a frequency of 50Hz. The grid must always maintain an equilibrium between supply and demand, otherwise there could be detrimental affects to equipment that assume the electricity remains at a constant frequency. Therefore, incorporating intermittent renewable energy that is distributed throughout the grid requires new methods of coordination. An additional challenge to this changing structure is the overall increasing demand for electricity, caused in part by the rising popularity of electric vehicles.

A current solution to balancing supply and demand is demand response. This is generally operated by a central entity, which is good for controlling the current grid, but will make it difficult to control a distributed one. Further research lead to the discovery of transactive energy and PowerMatcher, an open source transactive energy software project, as a potential solution [9]. PowerMatcher is a multi-agent system that coordinates multiple devices that are consuming and producing power. PowerMatcher has been in development since 2012 and piloted in numerous field tests. However, I observed that most field tests applied PowerMatcher in very specific scenarios. I realised the necessity to prove PowerMatcher's effectiveness in the broad-scoping issue of reducing a neighbourhood's grid usage by independently balancing supply and demand. These neighbourhoods would act as a microgrid, having their own supply and coordination but also needing the grid intermittently. When connected to the grid, it is important for the microgrid neighbourhood to reduce its peak to average demand ratio. Keeping this as low as possible would reduce the complexity in coordinating the main electricity grid.

> **Research Objective 1:** Investigate PowerMatcher's ability to balance the supply and demand of a modern neighbourhood consisting of renewable sources and electric vehicles.

In using PowerMatcher, I noticed an issue introduced by the charging of electric vehicles that caused a peak in demand. PowerMatcher delays the charging of the electric vehicles if there is no supply available in the system. However, the vehicles will eventually separate from the PowerMatcher system and consume from the main grid because they need to be charged by a certain time. I believe with added coordination this problem can be solved.

> **Research Objective 2:** Design an extension for PowerMatcher that will coordinate devices in order to reduce the peak to average demand of a system containing large loads.

In this dissertation, I become familiar with PowerMatcher, create simulations to test its functionality and design an extension for it. I have also created a demonstration using Raspberry Pi computers to control real electric loads.

1.2 Dissertation Structure

Chapter 2 provides background information to become familiar with the topics discussed in this dissertation. Chapter 3 gives an overview of PowerMatcher, how it functions, and important field tests that verify its benefits. In Chapter 4, I describe the design for the simulations and the extension to PowerMatcher. Chapter 5 covers the implementation of the scenario that prompted the design of the extension. It then discusses the technologies and challenges they introduce. Chapter 6 introduces future projects that could be built upon this one.

Chapter 2

Background Research

The War of Currents was a battle between alternating current (AC) and direct current (DC) electric power transmission systems in the late 18th Century [10]. The AC system eventually won this battle because AC electricity can be easily increased to higher voltages and transmitted over longer distances. This reduced costs as it was cheaper for a town to have one large centralised power generation plant than several smaller ones.

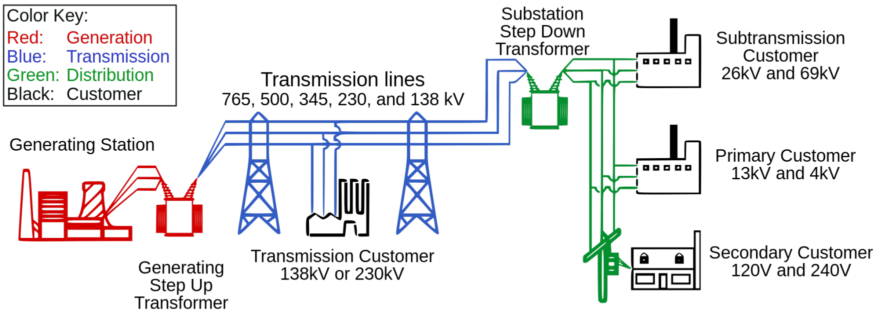


Figure 2.1: *A simple diagram showing the structure of the electricity grid [1]*

.

Although the electricity grid has become more complicated due to an increase in consumers, the grid structure has remained the same since the 18th Century. This structure, shown in Figure 2.1, provides reliable transmission and distribution of electricity. However, changes in the way electricity is produced and consumed are forcing the structure to evolve.

Namely, the grid is becoming decentralised as generation is distributed across multiple sources and there is a trend towards the electrification of everything.

The cause of these changes and the challenges they raise are given below. This chapter will introduce the terms described in Chapter 1: peak to average demand ratio (a metric used to measure the variability in electricity demand), demand response (the current method for reacting to increased demand and variable supply, and reducing peak to average demand), transactive energy (a new concept that is allowing demand response to become more automated), and microgrids (a small-scale version of the grid).

2.1 Generation Becoming Decentralised

The traditional electricity grid consists of large centralised fossil fuel power generation and a radial distribution network. In recent years, there has been a push for renewable energy sources, causing the grid structure to diversify. Large solar or wind farms still follow the traditional centralized structure, but there are increasing numbers of residential and municipal solar generation systems as well [11]. These fall under the category of distributed generation and are causing the grid to become more decentralised.

The push for renewable energy is caused by three factors: climate change concerns, depletion of fossil fuel reserves, and countries desires to diversify their energy sources.

Climate Change Concerns

Burning fossil fuels releases carbon dioxide into the atmosphere, causing rapid increases in ocean acidification and climate change [12]. 195 countries showed their consensus on this issue as they signed the Paris Agreement, stating that adopting renewable energy sources is key to battling climate change [13]. Renewable energy sources have been shown to be a viable alternative to fossil fuels and have a crucial role in mitigating climate change [14].

2.1.1 Depletion of Fossil Fuels

Fossil fuels are a finite resource. A team of researchers analysed changes in export tax for fossil fuels and estimated that oil will last for another 50 years, at most [15]. Although methods such as fracking for natural gas or shale oil extraction give access to more fossil fuel reserves, these methods have a lower energy-return-on-investment (EROI). A history of EROI shows that oil and gas have reached peak value for production, while coal is soon approaching its peak [16]. Therefore, it makes sense economically to divest from fossil fuels and invest in renewable technology.

2.1.2 Energy Diversification

Countries that depend on imported energy sources are realising that they can now use renewable sources to become more energy independent and to save financial resources. The European Commission estimates that renewable sources contributed to fossil fuel import savings of 16 billion in 2015, with savings projected to rise to 58 billion by 2030 [17].

2.1.3 Challenges Introduced by Decentralised Grids Powered by Renewable Sources

For these and many other reasons, renewable sources are quickly replacing the need for fossil fuels while simultaneously altering the way that energy is delivered to the home.

There are many challenges introduced by the changing structure of the grid. Here I focus on two major problems:

- **Indeterminacy:** Renewable energy sources have a stochastic behaviour. Weather predictions can be used to estimate power production, however the electricity generated may come at a time when it is not needed
- **Coordination:** Distributed generation increases the number of resources a grid operator must take into account. This adds a layer of complexity when coordinating the

electricity grid to balance supply and demand.

2.2 Growing Electricity Demand

Energy demand, and in particular electricity demand, is increasing. From 1990 to 2013, electricity production increased by 94% [18]. This increased demand is caused by an increase in population, in large due to population booms in developing countries, as well as an increase in the number of electric devices.

In the coming years, electric vehicles will raise this demand significantly. It is estimated that an electric vehicle will double an average household's energy consumption [19]. As governments increasingly incentivise and subsidise electric vehicles, they will become more commonplace. For example, nations such as France, China, and the United States of America have pledged to introduce more electric vehicles into their government fleet [20]. Germany, India and others have committed to stopping the sale of combustion engine cars by 2030 [21, 22].

Challenges Created with Increased Demand

While this shift in transportation will have broad benefits to the fight against the changing climate, it will also require new considerations and further energy adaptations. Two immediate and crucial challenges introduced by this increase in electricity demand are:

- **Supply and Demand Balance:** Addressing an increase in demand that will require more energy production. This is a huge challenge that will require new mechanisms for balancing supply and demand.
- **Grid Network Limitations:** The transmission and distribution lines of the electricity grid have a limited capacity. Simply producing more electricity will not solve the problem if it cannot be effectively transported.

The following sections discuss current technologies and research that aim to overcome the challenges mentioned here.

2.3 Peak to Average Demand

For an electricity grid to function, supply and demand must be in balance. This is important because if there is too much or too little supply to the grid then the frequency that electricity is delivered at will change, having adverse affects on household appliances and electronic equipment. As demand generally varies throughout the day, grid operators must constantly monitor demand and turn on or off power stations. This is much easier said than done, as power stations can take from minutes to hours to ramp up to appropriate generation levels. Generation on the grid consists of base load power stations such as coal or nuclear. These remain powered on continuously and supply enough electricity for the base load, the minimum demand on the system over a time period. When the demand increases above the base load, the grid operators will make use of intermediate power stations. In rare cases when the load increases to high points, the grid operator will need to use peaking power stations. These are normally gas power stations that can be turned on within 2 minutes.

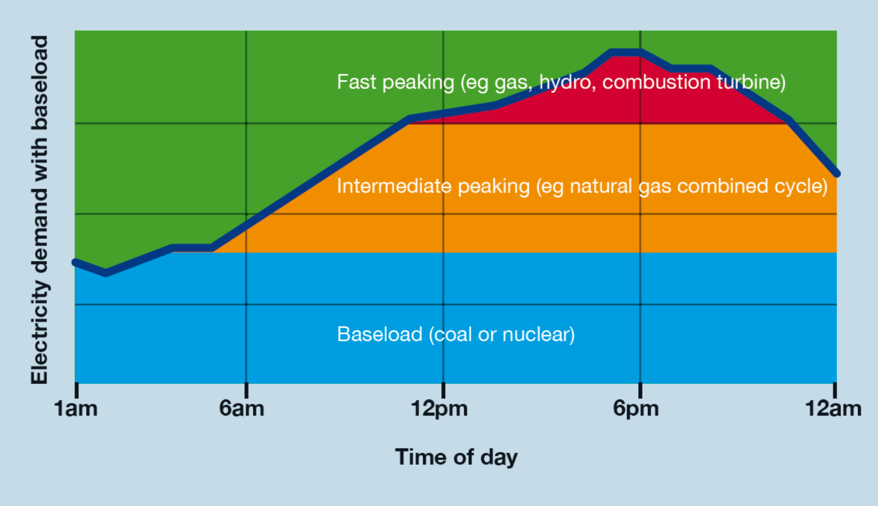


Figure 2.2: *An example of base, intermediate and peak loads [2].*

Figure 2.2 shows that because peaking plants are only utilised for a few hours a day, the electricity they generate is expensive, as operating costs are paid to keep the plant on standby. Making the electricity demand more predictable by eliminating the peak would make the coordination task easier and more economical, as peaking power stations would not be needed.

In an ideal situation, there would be constant demand that never changed. Thus, one of the goals for this project is to reduce the peak to average demand while using the electricity grid, avoiding all spikes in electricity demand.

2.4 Demand Response

Demand response (DR) ‘is a tariff established to motivate changes in electricity use by end-use customers in response to changes in the price of electricity over time, or to give incentive payments designed to induce lower electricity use at times of high market prices or when grid reliability is jeopardized’ [23]. In other words, it is the act of changing demand rather than supply to create an equilibrium in the electricity grid. Demand response is used widely today and a good solution to lowering the peak to average demand. There are multiple techniques for reducing demand that fall under the category of demand response. An overview of current techniques, advantages and disadvantages are discussed in detail in literature [24, 25, 26, 27]. Below is a summary of the current techniques that exist in the industry today and following this, I introduce a new method for automated demand response called transactive energy. Transactive energy is the basis for PowerMatcher communication and how demand response is performed in this project.

2.4.1 Demand Response Consumers

An important distinction to make in DR is between consumers. There are three main consumer types that participate in DR, as outlined by Vardakas et al. [25]. They are residential, commercial and industrial.

Commercial

Commercial DR customers tend to be universities or large office buildings. These facilities may have common usage patterns, such as an office building always being empty on the weekend. Managers of the building can take part in DR by turning off large air-conditioning systems or lighting systems at these times. Another possibility is that consumers will have on-site generation that they can alternative between to reduce demand on the grid. For example, a hospital or a data centre may have emergency back-up power.

Industrial

Industrial DR customers are normally the most effective, as they have large loads that can be used in DR such as manufacturing equipment or cooling systems. Very often, this equipment can be shut down for small periods of time with little cost to the business.

Residential

Residential DR customers provide the lowest potential peak reduction. Due to the small scale of individual customers, it can be difficult to coordinate and persuade residential customers to participate in DR. Additionally, it is challenging for a utility to make the savings for a residential customer worth the inconvenience of actively controlling small loads required by the customer. However, this could change as electric vehicles introduce a large load to residential customers as mentioned in Section 2.2.

2.4.2 Time Based Demand Response

The majority of residential, commercial and some industrial electricity customers will pay a fixed rate for electricity. This is an agreed upon rate with the utility. In reality though, the utility is paying different prices for the electricity they supply to the customer, depending on the current market cost [28].

Time based DR offers customers more than one set price, or varying prices for electricity in order to encourage the customer to use electricity at different times. There are three main DR programs that use a time based scheme. Those are Time Of Use (TOU), Critical Peak Pricing (CPP), and Real Time Pricing (RTP).

- **TOU** offers fixed rates for certain periods of the day. For example, Electric Ireland offers a ‘nightSaver meter’ [29]. This encourages the user to shift large loads to later in the night, away from normal peak demand time.
- **CPP** operates in a similar way to TOU, the difference being that rates can be changed a day ahead. This gives the utility more control and allows them to try and shift loads differently on specific days.
- **RTP** offers users dynamic pricing in real time, however it requires customers to react immediately to the price changes.

Time based DR can be successful at reducing peak demand, solving the grid network limitation challenge. However, it is an optional method and customers do not provide confirmation that they will be reducing demand. This means that the grid operator still cannot fully predict demand and must keep peaking plants available. Due to the manual reaction to price required by real-time based DR, it is generally not successful on residential scales. This is because residential electricity users do not have large loads to turn off so savings are minimal. However on a large industrial scale, customers can shift large loads such as manufacturing and cooling to cheaper times with little inconvenience, while gaining large cost savings on cheaper prices.

2.4.3 Incentive Based Demand Response

Incentive based DR is a method in which utilities or other DR providers will pay customers to enrol in their program. Customers will reduce loads according to the agreement made after being contacted manually or, in some cases, controlled remotely. The four most common methods of incentive DR are Direct Load Control (DLC), Emergency DR, Bidding DR, and Capacity Market DR.

- **DLC** gives control to a utility or DR provider, allowing them to remotely turn off loads at customer sites. There must be a device to allow control installed at the load. These loads are typically devices like water-heaters that can be switched off without causing any major inconvenience to the customer. The customer will receive a fixed payment for being enrolled in the program.
- **Emergency** DR programs pay customers based on how much demand they can reduce in times of grid emergency, i.e times of low supply. However, some emergency DR programs will also penalise customers if they do not reduce demand during emergency events.
- **Bidding** is a method that allows large consumers to bid their capacity on the wholesale market. Their capacity is how much they can reduce demand by. If successful in their bid, they are paid for reducing the demand.
- **Capacity Market** DR is similar to emergency DR in that customers are paid for reducing load during specific events. However, they are also given an upfront payment for committing to have a certain amount of demand available to reduce. For example, a manufacturing plant will have a machine that uses 100kW everyday and will commit to turning it off, should the utility require it.

Incentive based demand response is more useful in overcoming the supply and demand balancing problem. This is because utilities and grid operators can predict how much they can reduce demand by, not having to rely on consumers using periods of cheaper electricity. An issue with both time based and incentive based DR is that they both rely on centralised control. A central utility, DR provider or grid operator is responsible for predicting the need for a demand response event and implementing it. Although there can be some level of automation in DR using standards such as OpenADR to control remote loads, DR is mostly a manual procedure [30]. Automation is required to effectively respond to the variable supply that comes with distributed renewable sources. Transactive energy is a concept that aims to reduce peak demand and balance supply and demand through automated communication between loads.

2.5 Transactive Energy

The GridWise Architecture Council define transactive energy as ‘a system of economic and control mechanisms that allows the dynamic balance of supply and demand across the entire electrical infrastructure using value as a key operational parameter’ [31]. In a review of transactive energy, Chen et al. highlight the importance of distributed intelligent devices controlling loads and distributed energy sources [32]. These devices are controlled in real time and are managed automatically. The devices must make decisions based on economic incentives. The ‘value’ used could be as simple as the market price for electricity, or a more complicated value that incorporates user comfort. For example, a customer may place more value on their house being a comfortable temperature, or having their electric vehicle charged by a certain time.

2.5.1 State of the Art

Chen et al. provide a recent state of the art review on transactive energy [32]. However, it omits important demonstrations that have been included by Hammerstrom et al. and Kok et al. [33] [34].

The Pacific Northwest Smart Grid Demonstration Project

The Pacific Northwest Smart Grid Demonstration Project was a 5 year, large scale smart grid demonstration project [35]. The first transactive energy system was implemented as part of the project [36]. Feeder nodes on the distribution network allowed devices connected to it to trade on a novel market. The devices included electric water heaters, thermostats, battery storage, air conditioners and many others. Devices were traded on a 5 minute market with an aim of balancing supply and demand, reducing peak demand to stay below constraints, and adhering to customer comfort settings. The project was a success and paved the way for future transactive energy projects.

VOLTTRON

VOLTTRON is a United States Department of Energy open source platform [37]. VOLTTRON does not provide transactive energy, but provides a platform on which transactive energy applications can be built. It provides basic agents and drivers to communicate with devices, a web-based manangement interface to manage agents and applications, a message bus to allow communication between agents as well as communication protocol. The full description of VOLTTRON's features are outlined by Katipamula et al. [38]. VOLTTRON has been used in demonstration and commercial projects at a small scale [39].

Blockchain

Chen et al. highlight that one of the main challenges faced by transactive energy is security in system management [32]. Centralised control of real transactions by devices may not be widely accepted by customers. Blockchain technology provides a trust-less, decentralised and transparent marketplace that could be used for transactive energy. These means that customers do not have to place their trust in a central utility. This method has been proposed by Horta et al. and Mattila et al. [40] [41]. An implementation of sharing solar power between residents using Blockchain has been tested in New York [42, 43].

PowerMatcher

PowerMatcher is 'the major European transactive energy-based coordination mechanism' that provides transactive energy smart grid coordination [34]. Chapter 3 describes PowerMatcher's operation and lists field tests that show its functionality.

2.6 Microgrid

Distributed energy sources give way to microgrids. The United States of America Department of Energy defines a microgrid as ‘a group of interconnected loads and distributed energy resources within clearly defined electrical boundaries that acts as a single controllable entity with respect to the grid. A microgrid can connect and disconnect from the grid to enable it to operate in both grid-connected or island- mode’ [44]. Microgrids add resiliency to the grid, as they can operate independently of the main electricity grid. The ability to act independently allows renewable sources to be managed and coordinated on a small scale. This is because the grid operator can see the whole microgrid as one entity rather than several smaller sources and loads, increasing efficiency in load coordination.

Traditionally, microgrids are centrally controlled, often with manual switch over to the main grid [45]. However, recently automated approaches have been adopted, including research into transactive techniques for controlling microgrids [46].

2.7 Summary

This chapter has discussed factors that are causing the electricity grid to change, and why action must be taken to incorporate these changes. The grid is becoming decentralised and new methods of demand response, such as transactive energy techniques, will need to be used to balance the supply and demand. Separating the grid into microgrid structures can allow more fine grained coordination of resources. PowerMatcher is a coordination tool that uses transactive energy and could coordinate new microgrids.

The following chapter provides an overview of PowerMatcher and how it can be used.

Chapter 3

PowerMatcher

This chapter introduces PowerMatcher [47], outlines its basic functionality and provides some examples of PowerMatcher at work in the real world. It then identifies a problem with PowerMatcher and introduces an addition to PowerMatcher to provide a solution to that problem.

PowerMatcher is open-source software that is distributed by Flexiblepower Alliance Network that provides coordination for a smart grid [48]. ‘PowerMatcher is a “demand response” technology that balances all smart devices, from low voltage to high voltage, in a virtual market’ [3]. It is a multi-agent system comprising of three main agent types and a fourth optional agent type. Each energy source or energy consuming device is represented by a device agent, while the other three agents perform coordination of the system. These agents communicate using bid and price messages based on transactive energy techniques. The agents in PowerMatcher are structured in a hierarchy and can function in large scale networks [49].

Device agents communicate immediate energy needs for the current time slot in real time. While this means PowerMatcher can balance supply and demand in real time, it does not provide intelligent coordination of large future loads. As will be discussed in Chapter 4, electric vehicles are an example of large loads that are predictable to a certain degree. With current functioning capabilities, PowerMatcher will still cause large demand peaks if there is too much demand in a system. This danger could be reduced by adding two new message types for prediction and allocation of large loads, ensuring the system remains in balance.

3.1 Market-Based Control

PowerMatcher is a multi-agent system, which is defined as a collection of interacting autonomous agents [50]. As with most agent based systems, each agent has a goal. Designing this goal and ensuring that each agent achieves it correctly results in a successful system. In PowerMatcher, the goal is to consume or produce electricity at the best possible price. This is a method of market-based control. Market-based control is a design and implementation technique for regulating access to resources by putting those resources on a virtual or physical market and having individual agents compete for access to that resource. This is described in detail in by Kok [47]. Market-based control allows for a small number of agents to act unfairly or maliciously, assuming that the system is large enough so the malicious transactions on the market only affect the price fractionally. This assumption depends on the majority of agents to participate fairly.

3.2 Agent Structure

Agents are organised in a hierarchical tree structure as shown in Figure 3.1. The different agent functions are given in the documentation and are further described below [51].

3.2.1 Device Agent

Device agents sit at the bottom of the hierarchy. These agents represent actual devices such as electric vehicles, freezers, batteries and solar panels. The device agent software may run on a small computer such as Raspberry Pi and use General-Purpose input/output (GPIO) pins to communicate with the actual device. Alternatively, if it is a smart device, there may be an embedded system on the device that is capable of running the software.

The device agent coordinates with other devices by buying and selling electricity on a market created by PowerMatcher. The agent will send a bid curve to its parent agent, representing its desire to consume or produce electricity. The parent then returns a price to each device and who compares it to the previous bid to decide how much electricity can be utilised. After receiving the price, the device agent will construct a new bid curve and the cycle continues. Bids and prices are discussed in more detail below.

3.2.2 Concentrator Agent

The concentrator agent is in the middle of the hierarchy. These agents represent a cluster of device agents. Depending on the system, they may represent a room in a house, a whole building or an entire neighbourhood. The concentrator aggregates the bids made by its child agents and sends one bid for the whole cluster to its parent, usually the auctioneer agent, although there may be multiple levels of concentrator agents. Once it receives the price from the auctioneer agent, it communicates it back to all of the device agents below it.

The concentrator agent is useful for several reasons. The first, discussed in Section 3.5, is that it reduces the number of messages sent to the auctioneer which allows PowerMatcher to be scaled easily. Secondly, the concentrator can be used to allow more control of the cluster below it. For example, the concentrator agent could have knowledge of the current carrying capacity of the cables in its cluster. It can manipulate bids or prices to avoid these cables becoming overloaded. Finally, the concentrator can be used to add a layer of privacy to the system. By aggregating the bids of device agents, it obfuscates the individual bids of each device agent.

3.2.3 Auctioneer Agent

At the top of this hierarchy and at the centre of the PowerMatcher system is the auctioneer agent. The auctioneer aggregates the bids it receives from child agents and decides on the price. It then communicates this price back to all of the agents that sent a bid.

3.2.4 Objective Agents

PowerMatcher also describes an objective agent [48]. By default, a PowerMatcher system will aim to balance supply and demand. An objective agent can be added to the system to change that goal. In an example given by Kok, the system has to operate as a virtual power plant [47]. So at any given moment, there should be an excess of supply. The objective agent can supply false bids to insure the system reaches its goal.

Objective agents have not been used in this project as they have to be designed specifically for each scenario, whereas this project aims to extend PowerMatcher in a way that can be used generically.

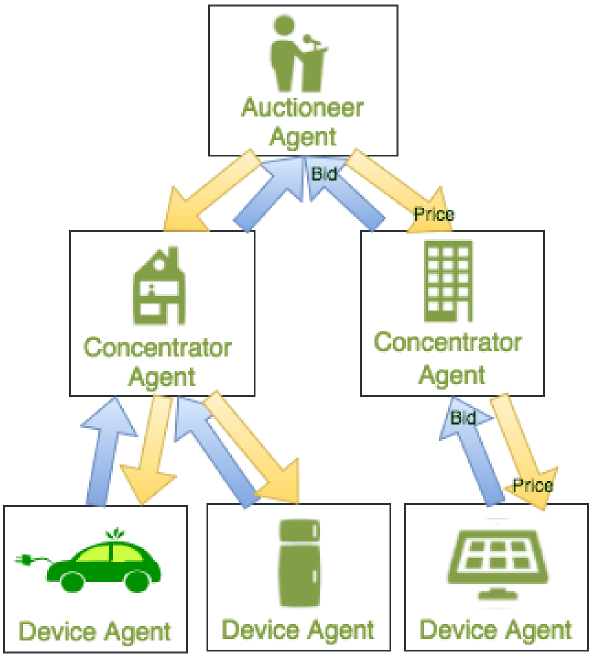


Figure 3.1: *The PowerMatcher hierarchy. Adapted from [3]*

3.3 Bids and Prices

All standard communication in PowerMatcher happens through bid and price messages. As mentioned previously, a bid is sent from a device and represents that device's desire to consume or produce electricity. The auctioneer looks at these bids and sends a price back to each device. The specifics of each of these messages is described below. It is possible that device agents are not associated with any concentrator and instead communicate directly with an auctioneer. However to make the explanation clearer I assume that a concentrator agent does exist.

3.3.1 Bid Curve

A bid curve is is a simple demand function that represents how much electricity a device wants to consume or produce. PowerMatcher formalises desire through price. The higher the price that a device agent is willing to pay for electricity, the more it desires it. In most scenarios and in this project, the cost of electricity is an arbitrary number. However, real price could be used to influence how agents behave, and this possibility is discussed in Section 6.1.1. As it represents this desire for a single time instance, the demand figure is in Watts.

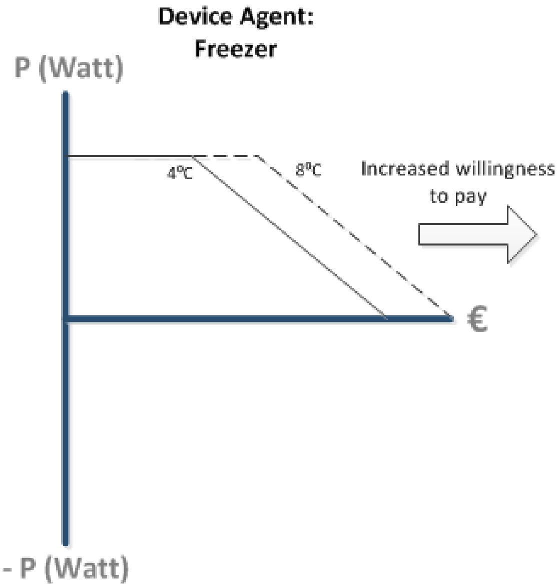


Figure 3.2: The bid curve for an example freezer. The x-axis represents the price the device is willing to pay for the corresponding power on the y-axis. As the temperature of the freezer increases, it is willing to pay more for electricity. Note that only one of the curves above is sent at a time, not both for 4° C and 8° C [4].

This is the core business logic of PowerMatcher: that each device will create its own bid curve resulting in agents competing for access to electricity. An example bid curve for a freezer is shown in Figure 3.2. Freezers allow for variance in temperature: assume this particular freezer can vary from 0°C to 10°C. If the temperature is at 0°C, the freezer will send a bid indicating that it does not want to consume electricity because it needs to get warmer. On the other hand, if the temperature is at 10°C, it will send a bid indicating that it must use electricity. If the freezer is at 4°C (shown by the solid line in Figure 3.2), it does not have an urgent need to consume electricity, but if electricity is very cheap at the moment it should take advantage of this. Therefore, a bid can be sent that indicates the device will consume electricity provided it is below a certain price point. It follows then that as the temperature of the freezer increases, so does its desire to consume electricity. This is shown by the dotted line in Figure 3.2.

A bid for a device that wants to produce electricity will simply have the demand value as negative. This is demonstrated in the bid curve made by a battery shown in Figure 3.3.

3.3.2 Bid Aggregation

Device agents send their bids to the concentrator agent they are associated with. Assume a concentrator agent has two child agents, a battery and a freezer. Figure 3.3 shows the bid curves for these devices. The battery sets a minimum price that it is willing to produce electricity for, shown by the gap at the beginning of its bid curve. It is likely that the battery's bid curve is a function of its current charge level.

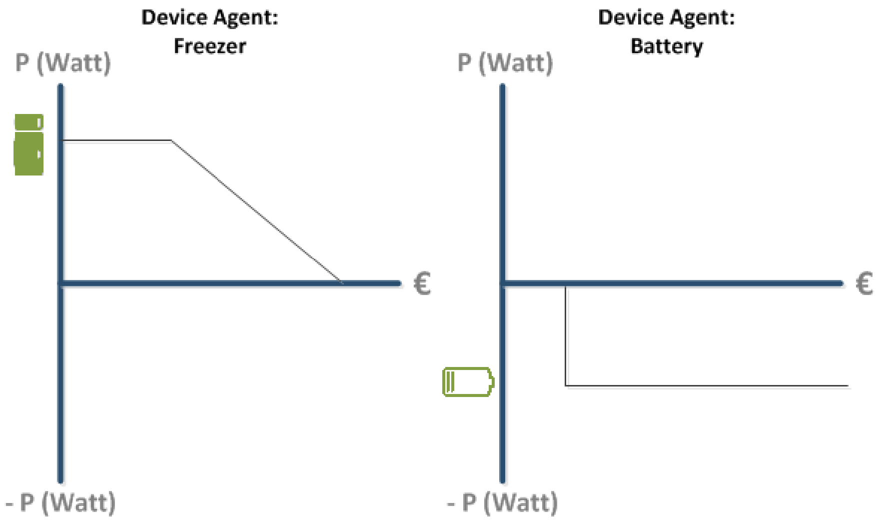


Figure 3.3: *The bid curve for a freezer and a battery. Note the battery has a negative bid as it is willing to produce electricity. [5]*

.

The concentrator agent aggregates these bids so that it is left with one bid curve that represents its child devices' as shown in Figure 3.4.

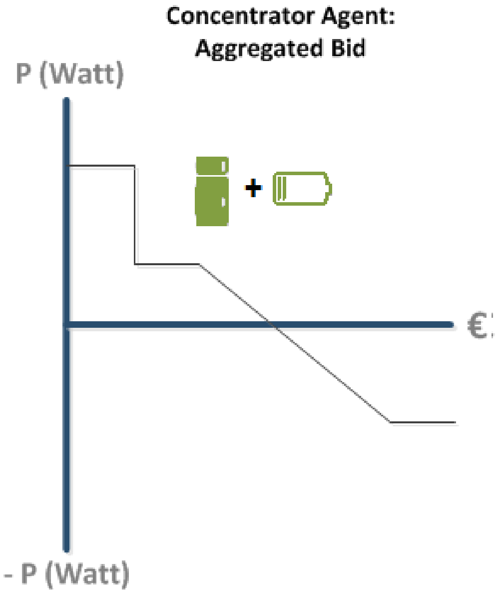


Figure 3.4: *The resulting aggregated bid curve produced by the concentrator by combining the bids given in Fig.3.3. [5]*

.

3.3.3 Equilibrium Price

The price that has been referred to so far is actually called the equilibrium price. This is the price that will leave the supply and demand of the system in equilibrium. Once the auctioneer agent receives all of the bids from the concentrator agents, it performs a final aggregation resulting in one bid curve for the whole system. The price point at which the final bid curve crosses the x-axis is the equilibrium price. This price is then communicated back to each device agent. The device agent finds the demand value for the given equilibrium and sets the actual device to the value. This process is shown in Figure 3.5.

It is possible that no equilibrium price is found and the resulting system is unbalanced. Depending on the implementation, this may be handled by forcing certain devices to power off or consume electricity from the electricity grid.

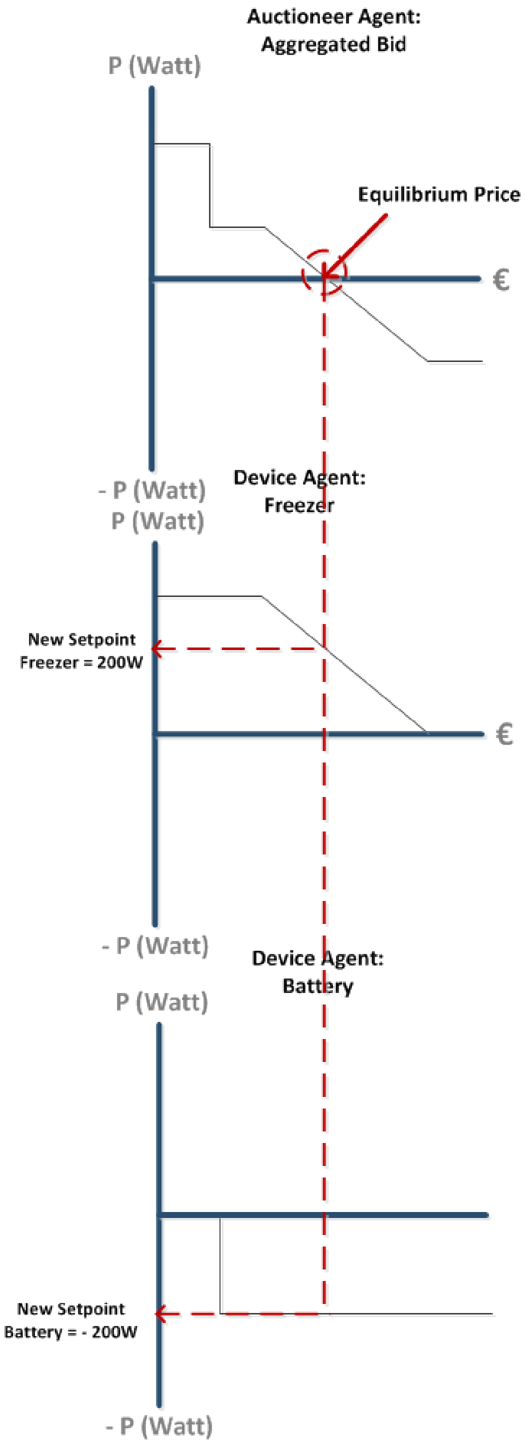


Figure 3.5: *The Equilibrium price calculated from the example battery and freezer. [6]*

.

3.4 Sessions and Device Communication

PowerMatcher uses the concept of a session to connect devices to each other. This is an implementation detail that means that no two devices are actually connected directly to each other. Instead, a session is created between them and both devices connect to this session. A lot of complexity can be handled by sessions to manage errors when one agent goes off-line or wishes to create new connections. The sessions also abstract the communication so any two agents can connect. For example, the concentrator appears as the auctioneer agent to the devices below it, but appears as a device agent to the auctioneer above it.

Sessions also abstract the complexity of wireless communication in a distributed PowerMatcher system. PowerMatcher uses WebSockets to allow devices to communicate wirelessly [52]. On the same computer as the device agent, a WebSocket client is created and set as the device agents parent. The WebSocket client is configured to connect to a corresponding server located on the parent agent that the device wants to connect to. The session is replicated by the client and server, making the connection transparent to the device and concentrator agents.

3.5 Scalability

One of the most important objectives for PowerMatcher is scalability. It is important that the system can scale to a suitable size so that it can be functional in the real world. PowerMatcher achieves this by eliminating a central control unit. Although there is a central node, the auctioneer agent, it does not have control over or direct contact with any of the device agents in the system. It is also favourable that any agent in the system can assume the role of auctioneer.

3.5.1 Message Size and Volume

The only aspect of PowerMatcher that can effect scale is the message design. If a network gets congested with large messages or an agent receives too many messages at once, the system may fail. In order to keep track of which agent sent which bid, auctioneer and concentrator agents must also keep a cache of recent messages received. As the target platforms for PowerMatcher are small embedded systems, this cache cannot grow too large.

An in-depth analysis of PowerMatcher's resource requirements concluded that large scale clusters of PowerMatcher are feasible using the existing communication technology [53]. It identifies three characteristics that make PowerMatcher communication feasible. The first is the ability to aggregate bids by intermediate agents in order to reduce message volume. The second is that each agent only has to publish its message once. Thirdly, each message sent is small in size which keeps the agent cache small.

It follows that in order to maintain scalability when extending PowerMatcher with new message types, they must adhere to those characteristics.

3.6 Field Tests

PowerMatcher capabilities have been verified in several field tests as summarised by Kok et al. and highlighted below [19].

3.6.1 PowerMatching City

PowerMatching City is a living lab smart grid demonstration [54]. It has undergone multiple iterations expanding on each previous form. Currently, it consists of 40 households in the Netherlands [55]. There are several experiments in effect there aiming to supply proof that the smart grid can support flexibility.

PowerMatcher was added during the second iteration. It was used by 22 households, each using PV panels, heat-pumps, smart washing machines and dishwashers. There were two electric vehicles in the system, one 5kWh battery and a 2.5MW wind turbine. The goal of the experiment was to show that PowerMatcher could handle the technical coordination, commercial coordination and home coordination of this system. The commercial coordination requirement was that the PowerMatching city acted as a virtual power plant and supplied electricity to the main electricity grid.

PowerMatching City was successful in showing that homes could handle the imbalance of generation from the wind turbine and reduce the load on the electricity grid.

3.6.2 Scalability Test

To verify the capability of PowerMatcher on a large scale, a field test was run with one million households. The households all acted as one virtual power plant operating in the balancing market. This is a market associated with the main electricity grid in which short term settlements are made to balance the system. In this case, the households bought electricity from the balancing market when there was excess production. The significance of acting on the balancing market is that it requires quick response times. The virtual power plant set a goal of reacting to an imbalance in the market within 5 minutes.

PowerMatcher was successful in coordinating the one million homes so that they could participate in the balancing market, verifying that it is scalable while maintaining low latency.

3.6.3 Wind Integration

As mentioned previously, one of the biggest barriers for renewable energy sources is that they are intermittent, forcing electricity grid operators to put a ceiling on the amount of electricity that can be produced by renewables. If demand reaches this ceiling, the electricity must be curtailed.

In a study by the SmartHouse project, PowerMatcher was used to handle the variability of large wind generation [56]. This was done in a large simulation consisting of 3000 homes that contained heating systems that could react to wind generation. It found that in using

PowerMatcher to coordinate these heating systems, there was an increased utilisation of wind generation. The simulation was run using several different capacity wind farms, the full results of are documented by Kok et al. [19].

3.6.4 Electric Vehicle Charging

The Flexiblepower Alliance Network also recognise that electric vehicles are soon to be a difficulty for the electricity grid. During a simulation based on a German mobility survey, PowerMatcher was used to coordinate the charging of electric vehicles. The goal of this study was to lower the peak demand of an average household using an electric vehicle. Without coordination, cars charge as soon as they are plugged in, normally when consumers return from work at about 6pm. This is the same time as the peak demand of the household. The simulation assumed the driver had assigned a time that the car needed to be charged by. Using this assumption, PowerMatcher could shift the demand of the electric vehicles to later in the night, lowering the peak to average demand ratio of the household.

3.7 Extending PowerMatcher

As the field tests show, PowerMatcher is successful at providing real time coordination of the smart grid. However, there is improvement to be made with loads that are predictable. In the last field test described in Section 3.6.4, the electric vehicles avoided the main peak load period of 6pm. As a result however, they created their own peak load later in the night. This second peak load may still cause problems for the electricity grid operator.

Continuing with the electric vehicle example, it is apparent that there is information available that is not being used. Each vehicle has information for what time it needs to be charged by, its capacity and its charging rate. The latest time the vehicle can start charging can be deduced from this. For example, the vehicle needs to be charged by 7:00, has 50%

charged of a 24kWh battery and charges at 3000W.

$$\begin{aligned} 24000Wh \times 50\% = 12,000Wh \\ 12,000Wh/3000W = 4hours \end{aligned} \tag{3.1}$$

Equation 3.1 shows that a device agent receiving this information can calculate that the latest the car can begin charging is 3:00. In other words, if PowerMatcher keeps raising the price above the electric vehicles set point, it will need to turn on at 3:00 regardless of the state of the system. If there are multiple vehicles that take the same approach then there will be a large peak at 3:00. By adding a small amount of prediction to PowerMatcher, this peak can be avoided.

3.7.1 Two-time Scale PowerMatcher

A similar approach was tested in the 'two-time scale' PowerMatcher presented by Kempker et al. [57]. The authors suggest a PowerMatcher system that operates on two time scales: a long-term scale for planning, and a short term scale for current operation. Each agent begins by estimating its demand curve over a long period, e.g 24 hours. This demand curve is based on the predicted electricity generation of the system, the devices predicted demand, and possibly information about previous demand curves. The auctioneer calculates a long-term price which each device agent then uses to calculate its short-term bid.

I propose a similar approach, however mine uses two new message types: prediction and allocation. I also assume the device agent does not have access to predictions about future electricity generation as described in the 'two-time scale' PowerMatcher. The prediction and allocation messages are also intended as an addition to PowerMatcher, to act as a back-up planning module if the normal PowerMatcher coordination causes an unexpected peak. This contrasts the 'two-time scale' PowerMatcher which uses the prediction to calculate how an agent should behave in the standard PowerMatcher.

In the next chapter I give my design for the prediction extension for PowerMatcher.

Chapter 4

Design

The need for the prediction extension to PowerMatcher became apparent while simulating a modern neighbourhood scenario. In this chapter, I describe this neighbourhood and outline how each device in the scenario is designed. Following this, the prediction and allocation messages are defined and I present an example of their usage capabilities in the neighbourhood scenario. I then explain the design approach of the bidding strategy for each device and conclude by outlining a plan for how this scenario could be realised using Raspberry Pi computers.

4.1 Modern Neighbourhood

A new type of neighbourhood is emerging. Soon, everyone will have an electric vehicle, solar panels and neighbourhoods may even share a wind turbine. With the adoption of the Internet of Things, devices are becoming smarter and capable of participating in the smart grid. In order to get a better understanding of how PowerMatcher operates, I wanted to simulate this modern neighbourhood and test PowerMatcher's ability to coordinate it.

During the unveiling of the Tesla solar roof, Elon Musk presented a neighbourhood similar to the one I have described [58]. Each house has an electric vehicle, a battery and a solar roof. This model made it clear that these three devices were probable to become common so

I chose to simulate them for my test.

4.2 Simulation of Devices

The device simulations below were designed so they could be easily configured with unique parameters. They were each designed to give an accurate but high level representation of the general functionality of these devices. This means that some details such as battery degradation and loss of energy transmission are not simulated.

4.2.1 Electric Vehicle

Electric vehicles are the main focus in this project. They are used to demonstrate the effect that widespread electrification will have on the electricity grid. Three models of electric vehicle were simulated based on data obtained from research conducted by Shao et al. [59]. The authors of this study provide battery capacity and charging power of the Tesla Model S, the Nissan Leaf, and the GM Chevrolet Volt, each of which are fully electric and in production.

The time the vehicle arrives home is chosen randomly between two times that are configured during the simulation initialisation. Similarly, the time the car needs to be fully charged by is randomly chosen between a different set of time bounds.

In future iterations of this simulation, the vehicle should be designed to allow different charging rates.

4.2.2 Solar Photovoltaic Panel and Wind Turbine

To simulate both solar and wind power, data was acquired using an API provided by renewables.ninja [60]. The API collects data from NASA's MERRA satellite and CM-SAF's SARAH dataset and converts them to power output using software developed by Pfenninger et al. [61]. The API provides data based on location and whether or not the solar panel or wind turbine has sophisticated tracking mechanisms. The solar panels and wind turbine in the simulation were designed so that different capacity systems could be easily added.

4.2.3 Battery Simulation

Usually, homeowners who install solar panels also install a battery. This is necessary so they can utilise the power generated by their solar panel in the middle of the night. The battery in this simulation was designed using the specification of the Tesla Powerwall [62]. For the purpose of the simulation, the battery was only able to produce and consume at fixed rates. This restriction was added to simplify the bidding strategy as explained in Section 4.3.2. For future iterations, the battery would produce and consume at any rate up to a limit.

4.3 Design of Bidding Strategy

With the simulation of each device designed, I could start working with the PowerMatcher software. Each simulated device requires its own device agent. The device agent will communicate with the other device agents based on the status of its simulated device.

For my implementation of PowerMatcher, the price an agent bids is arbitrary. Each agent makes a bid that is a percentage of the maximum price of €1.

The developers of PowerMatcher, the Flexiblepower Alliance Network, have a software platform called The Energy Flexibility Platform and Interface (EF-PI) [63]. EF-PI has a similar concept of device agents as PowerMatcher does. In EF-PI, there are four different categories for device agents: Uncontrollable, Time Shift-able, Buffer/Storage and Unconstrained. All devices can fall under one of these categories and so the bidding strategies for devices in the same category will be similar to one other. I have designed the bidding strategies of each agent keeping these categories in mind.

4.3.1 Electric Vehicle

An electric vehicle could technically fall under two categories, depending on the implementation and capabilities of the vehicle. A concept called vehicle-to-grid systems will allow electric vehicles to provide energy back to the electricity grid, making it a buffer/storage agent [64]. However, I have not allowed for that functionality and instead designed the electric vehicle as a time shift-able device. These are devices that have a demand that can be postponed, started and stopped multiple times, but it must be fully charged by a certain time.

When constructing a bid, the device agent calculates a time-to-charge ratio. This is the ratio of time it will take the car to become fully charged to time until it needs to be finished charging. In the same example in Section 3.7, we found that a vehicle with a 50% charged 24kWh battery, charging at 3000W takes 4 hours to become fully charged.

If it needs to be fully charged in 12 hours time, then the ratio will be $1 : 3$ ($4hours : 12hours$). The device agent represents this ratio by constructing a bid that shows the agent is willing to pay up to €0.33 for 3000W. In 6 hours time, the ratio will be $1 : 1.5$ ($4hours : 6hours$ and the vehicle will be willing to pay €0.66 for electricity. This progression is shown in Figure 4.1.

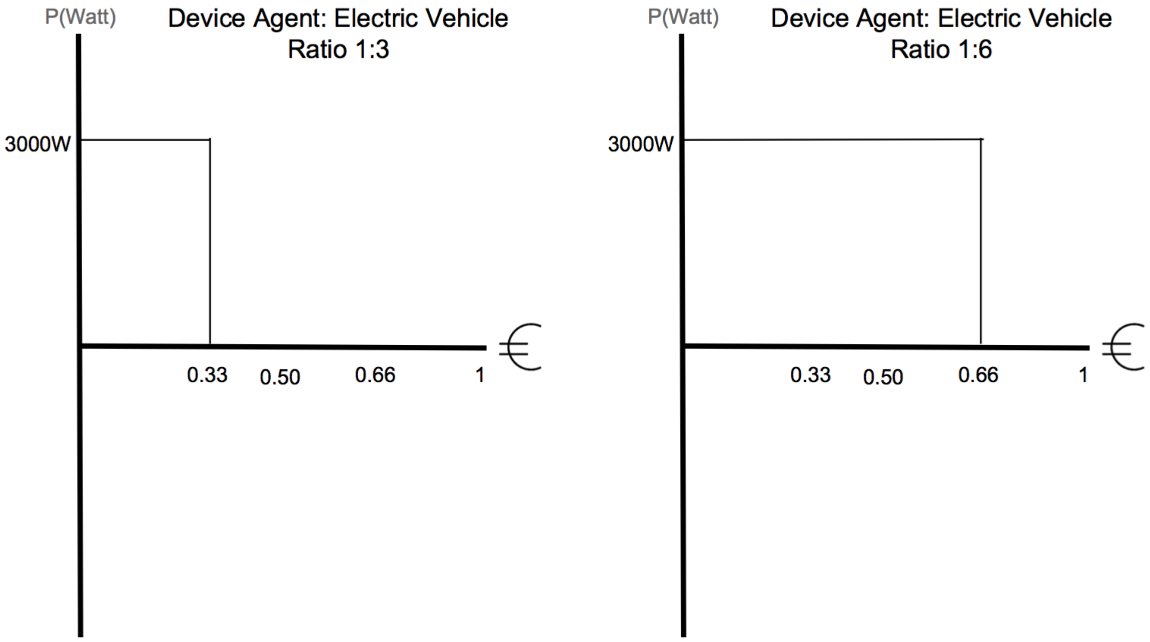


Figure 4.1: *The bidding strategy of an electric vehicle. On the left, the vehicle has a time to charge ratio of* $1:3$, *on the right the ratio is* $1:1.5$.

This design only allows electric vehicles to charge at a fixed rate. Future iterations will have a more complicated bidding strategy that allows the vehicle to charge at varying power rates.

4.3.2 Battery

A battery falls under the EF-PI category of storage device. The Device agent for a battery needs to find a price at which it will buy electricity, and a price at which it wants to sell electricity. Economically, the more charge a battery has, the more it will want to produce electricity. However as a battery loses it charge, it will start offering to buy electricity. It follows that the Device agent calculates a bid that is a function of the percentage charged. For example, if the battery is 25% charged, it will make a bid to consume electricity up to €0.75. However, if the equilibrium price is over this bid, the battery will produce electricity.

4.3.3 Solar Photovoltaic Panel and Wind Turbine

Solar panels and wind turbines are both uncontrollable devices. No one can control the sun shining or the wind blowing so there is no flexibility with these devices. Uncontrollable devices produce a flat bid for the current power value they are producing. This type of bid means that they will produce electricity at a specified power, regardless of the equilibrium price.

4.4 The Scenario During Normal PowerMatcher Operation

The above simulations and bidding strategies were designed to observe PowerMatcher's behaviour when operating in the given neighbourhood scenario. During the day, the solar panels charge the batteries and in the evening, the electric vehicles arrive home. PowerMatcher appropriatly balances supply and demand by charging some of the vehicles to make use of the little solar energy that is left. It then allows vehicles to charge from the batteries, however home batteries do not have enough capacity to fully charge the vehicles. As the PowerMatcher system is just in control of these devices, at the end of the evening it cannot balance the system as there is no supply left but there is still the demand of the electric vehicles.

The results of this simulation proved PowerMatcher to be successful, as it has reduced the peak demand that would occur once all the vehicles are plugged in. The issue remains that the vehicles still need to charge. PowerMatcher can only postpone the vehicles a certain amount of time until they ignore the price signals and charge using the electricity grid. This causes a second peak to occur in the middle of the night. Observing this phenomenon, I noticed a simple prediction mechanism could give PowerMatcher control of the second peak created.

4.5 Prediction and Allocation Message Design

I designed two new message types to use in PowerMatcher. The goal of these messages is to reduce peak demand created by multiple predictable large loads. The theory is that certain devices, such as electric vehicles, can predict the time they will need to use electricity if they have not been given permission earlier. Given this information from several devices, an auctioneer agent can shift some of these large loads so that they are not all consuming electricity simultaneously, thus reducing the peak demand.

To maintain the scalability of PowerMatcher described in Section 3.5, there were three requirements the new messages must meet:

- **Small message size** - The size of the messages must not greatly exceed the size of the bid message.
- **Aggregable** - It must be possible to easily aggregate multiple prediction messages.
- **Universality** - The auctioneer should only have to send one message to its child agents, not different messages for each agent.

4.5.1 Prediction Message Calculation

The first message added is the prediction message. This message is used to give a prediction of an agents demand over a time period determined by the specific implementation of the prediction message. For the purpose of this explanation, assume that the time period is 12 hours.

To predict the device's load, the device agent must have some previous knowledge such as a regular schedule to make the prediction simple. In the case of an electric vehicle, the device agent must know what time the vehicle needs to be charged by. This example has been discussed in Section 3.7. Equation 3.1 found that if the car was not given any supply from the PowerMatcher system, it would charge from the main electricity grid at 3:00. The device agent makes this calculation and creates a prediction. The prediction message for this vehicle is shown in Figure 4.2.

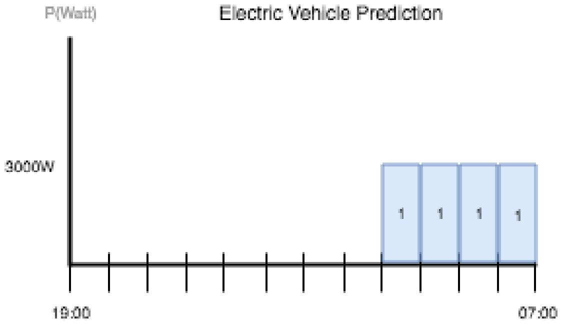


Figure 4.2: *The prediction message for an electric vehicle. The colour represents the agent's ID. The number represents how divisible the demand value is.*

The prediction message contains three attributes for each time slot. The height of each bar is the demand required for the slot. The colour/shading is the ID of the agent. Finally, the number represents how many times that predicted load can be divided. In the case of the electric vehicle, it is 1 as it cannot divide its load. Divisibility becomes important at the aggregation step and will be discussed in detail below.

Aggregation of Prediction Messages

When a concentrator or auctioneer agent receives multiple prediction messages, it aggregates them by summing the demand values and the divisibility factor for each time slot. Assume the Concentrator receives another message from an electric vehicle with a predicted load between 4:00 and 6:00. The resultant message is shown on the left of Figure 4.3.

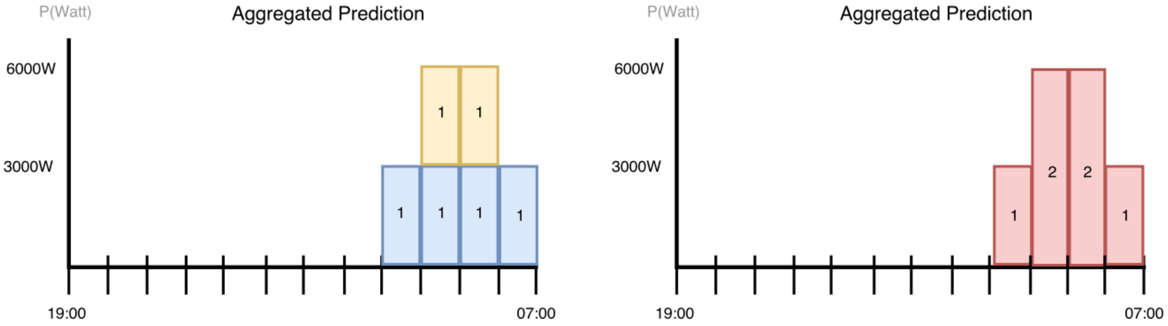


Figure 4.3: *The prediction messages received by a concentrator on the left, and the message it sends on the right.*

On the right of Figure 4.3 is the message the concentrator will send to the auctioneer. It has replaced the ID of the electric vehicles with its own ID. It must cache the original IDs of the electric vehicles for use later.

4.5.2 Allocation Message

Upon receiving the prediction messages, the auctioneer begins the allocation process. The auctioneer has the aim of reducing the peak as much as possible and does this by dividing demand values if possible. In this example, the demand between 4:00 and 6:00 can be divided in two. The agent distributes the halves to time slots that are free: in this case, between 2:00 and 4:00 as shown in Figure 4.4.

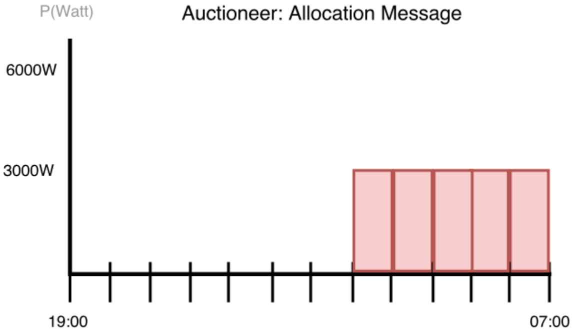


Figure 4.4: *The auctioneer creates an allocation message by shifting the peak loads.*

The auctioneer can send this message to all of the agents it receives prediction messages from.

Distributing Allocation Message to Original Devices

The final step in allocation is to distribute the messages. The concentrator must reassign IDs to the demand values for each time slot. The concentrator must look in its cache to match demand values of the same size. This computation step is similar to the computation required during price messages. Price messages are assigned the ID of the bid they correspond to.

The device agent receives the final allocation message and begins consuming electricity during its new allocated time slot.

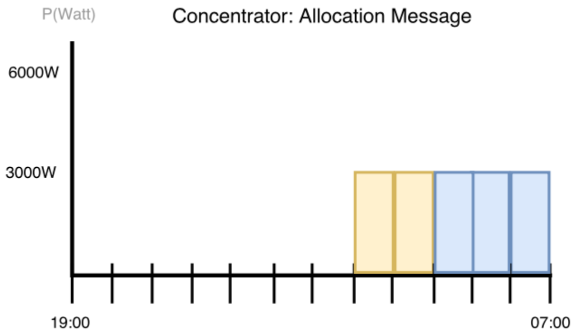


Figure 4.5: *The concentrator must reassign the original device agent IDs*

4.5.3 Frequency of Messages

As with the frequency of bid messages, the frequency of prediction messages depends on the system design. If the time slots in the prediction message are 1 hour each, then the prediction message should not be sent more than once an hour. This is to allow the allocation to become stable and propagate through the network.

4.6 Raspberry Pi

PowerMatcher is designed to run on small computers such as a Raspberry Pi or Arduino. As I had access to Raspberry Pi devices, I chose to use it to simulate the operation of the system. Although PowerMatcher can be fully simulated on one device, I wanted to prove the functionality of the system in the real world.

My original design incorporated multiple Raspberry Pi computers, each representing a single house. The Raspberry Pi would act as the controller for that house, connecting multiple loads to multiple sources. The controller would also act as a concentrator agent which would then communicate with the rest of the system. One of these controllers could also act as the auctioneer for the whole system. Each device agent would be implemented and run independently on the Raspberry Pi controller.

Raspberry Pi's can be extended using different modules. One of these is a relay board. The relay board allows for the turning on and off of devices as well as switching between power sources.

To simulate real loads, such as the electric vehicle, lights were connected to the relays. There are two energy sources connected: the main electricity grid and a battery that is charged by a solar panel.

4.7 Summary

This chapter outlines the design of the simulations used in this research. Four types of device are simulated: electric vehicle, solar PV panel, wind turbine and battery. The design of the PowerMatcher device agents for each of this device types is also given here. The extension to PowerMatcher proposed in this research is a prediction mechanism. This extension requires two new message types: prediction and allocation. The functionality and design of these messages is also given in this chapter.

The following chapter provides the details of the simulation that prompted the design of the extension. It also describes the Raspberry Pi demonstration. Finally it gives details on challenges faced while developing with the technologies used by PowerMatcher.

Chapter 5

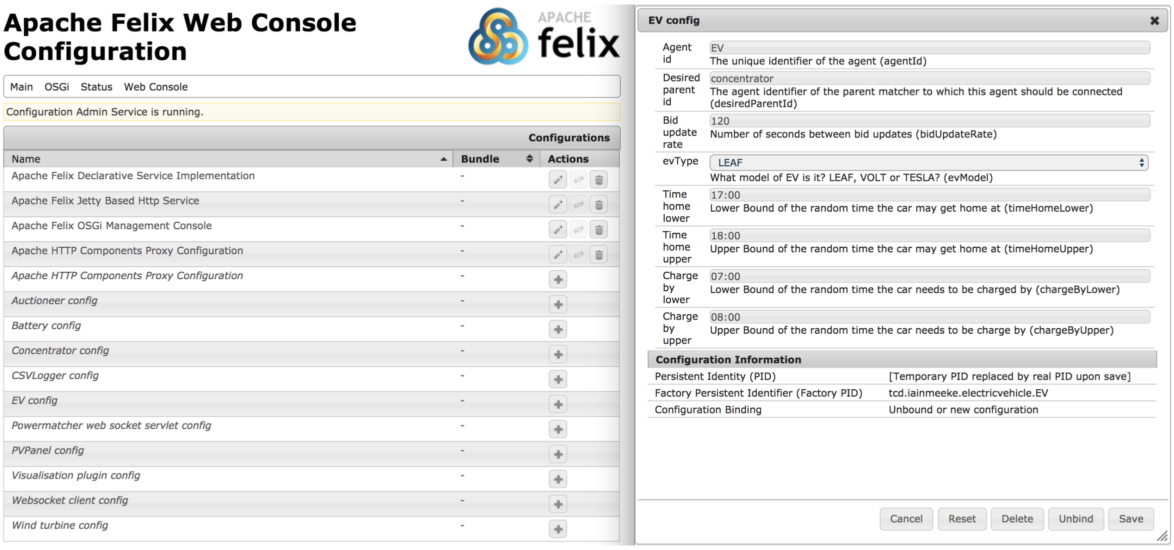
Implementation Details

PowerMatcher is a complex piece of software. Before developing device agents or simulations, it was necessary to become familiar with the PowerMatcher code base and technologies used.

This chapter gives the process for running an example simulation and also describes in detail some of the issues encountered while using PowerMatcher.

5.1 Implementing A Simulation

In this section, I outline the basic simulation in which I observed the behaviour that caused PowerMatcher to create demand peaks by shifting electric vehicle loads. In this simulation, a vehicle is controlled by a PowerMatcher auctioneer agent. As will be discussed in Section 5.3, PowerMatcher runs in the Apache Felix OSGi container. PowerMatcher also makes use of the Apache Felix Web Console. Through the Web Console device agents can be added to the system. The Web Console used in this project is shown in Figure 5.1. The list of available agents is shown on the left. The configuration options for an electric vehicle agent are shown on the right. In this project, adding an agent also starts a simulation for the agent to control. The vehicles in this example arrive home and are plugged in between 17:00 and 18:00 each day. The vehicles are required to be charged between 08:00 and 09:00. The specific time is different on each day. Without PowerMatcher, the vehicles would begin charging as soon as they were plugged in. This is during the normal peak demand time. As there are no other devices, and therefore no supply for PowerMatcher to control, the electric vehicle charging gets delayed as long as possible. This behaviour is shown in Figure 5.2. The vehicle begins charging at the latest time possible.



Apache Felix Web Console Configuration

Configuration Admin Service is running.

Configurations

Name	Bundle	Actions
Apache Felix Declarative Service Implementation	-	
Apache Felix Jetty Based Http Service	-	
Apache Felix OSGi Management Console	-	
Apache HTTP Components Proxy Configuration	-	
Apache HTTP Components Proxy Configuration	-	+
Auctioneer config	-	+
Battery config	-	+
Concentrator config	-	+
CSVLogger config	-	+
EV config	-	+
Powermatcher web socket servlet config	-	+
PVPanel config	-	+
Visualisation plugin config	-	+
Websocket client config	-	+
Wind turbine config	-	+

EV config

Agent id	EV The unique identifier of the agent (agentId)
Desired parent id	concentrator The agent identifier of the parent matcher to which this agent should be connected (desiredParentId)
Bid update rate	120 Number of seconds between bid updates (bidUpdateRate)
evType	LEAF What model of EV is it? LEAF, VOLT or TESLA? (evModel)
Time home lower	17:00 Lower Bound of the random time the car may get home at (timeHomeLower)
Time home upper	18:00 Upper Bound of the random time the car may get home at (timeHomeUpper)
Charge by lower	07:00 Lower Bound of the random time the car needs to be charged by (chargeByLower)
Charge by upper	08:00 Upper Bound of the random time the car needs to be charge by (chargeByUpper)

Configuration Information

Persistent Identity (PID)	[Temporary PID replaced by real PID upon save]
Factory Persistent Identifier (Factory PID)	tcd.iainmeeke.electricvehicle.EV
Configuration Binding	Unbound or new configuration

Figure 5.1: *The Felix Web Console for configuring Agents. On the left is the list of available bundles (see Section 5.3), on the right are the configuration options for an electric vehicle.*

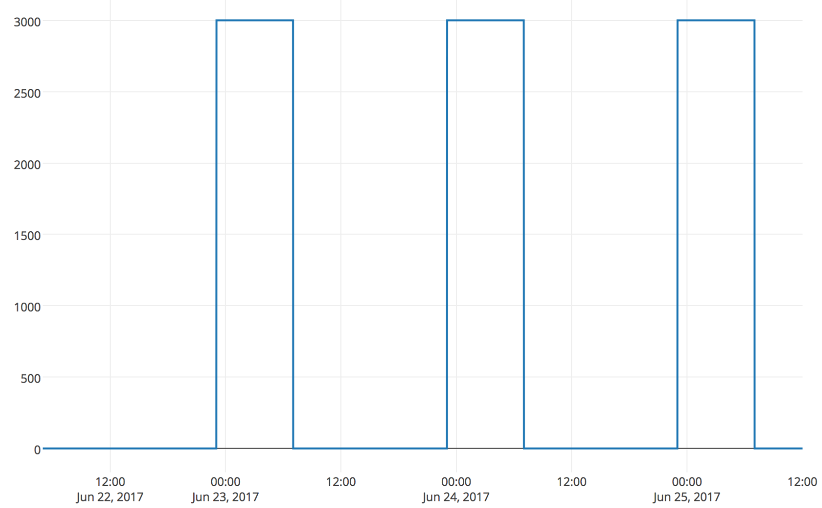


Figure 5.2: *The demand for the electric vehicle that has its charging postponed to later in the night.*

Adding a solar PV panel to the system shows that PowerMatcher can control the timing of charging to make use of the solar power that is still available when the vehicle arrives home. This behaviour is shown in Figure 5.3. The configuration of the solar PV panel is shown in Figure 5.4.

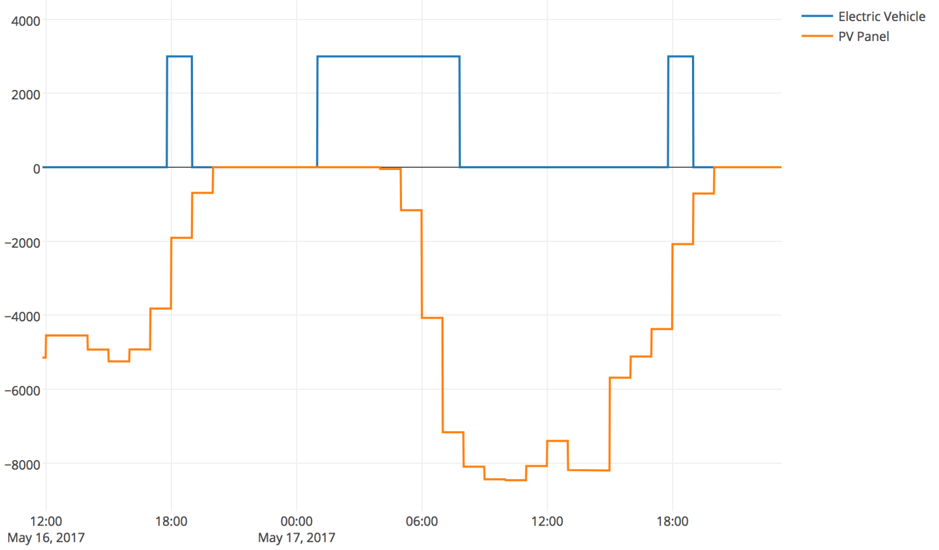
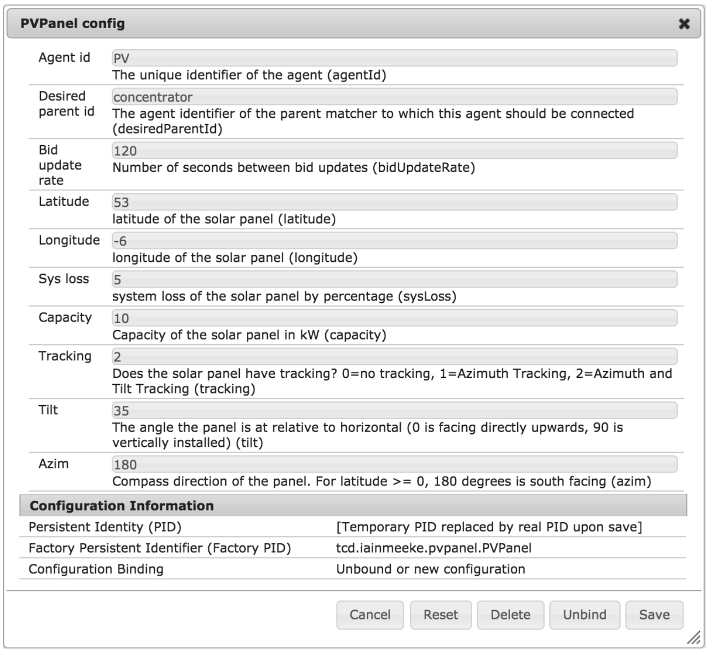


Figure 5.3: *The demand for the electric vehicle and the output for a solar PV panel.*



PVPanel config

Agent id	PV The unique identifier of the agent (agentId)
Desired parent id	concentrator The agent identifier of the parent matcher to which this agent should be connected (desiredParentId)
Bid update rate	120 Number of seconds between bid updates (bidUpdateRate)
Latitude	53 latitude of the solar panel (latitude)
Longitude	-6 longitude of the solar panel (longitude)
Sys loss	5 system loss of the solar panel by percentage (sysLoss)
Capacity	10 Capacity of the solar panel in kW (capacity)
Tracking	2 Does the solar panel have tracking? 0=no tracking, 1=Azimuth Tracking, 2=Azimuth and Tilt Tracking (tracking)
Tilt	35 The angle the panel is at relative to horizontal (0 is facing directly upwards, 90 is vertically installed) (tilt)
Azim	180 Compass direction of the panel. For latitude >= 0, 180 degrees is south facing (azim)

Configuration Information

Persistent Identity (PID)	[Temporary PID replaced by real PID upon save]
Factory Persistent Identifier (Factory PID)	tcd.iainmeeke.pvpanel.PVPanel
Configuration Binding	Unbound or new configuration

Figure 5.4: *The configuration for the solar PV panel and simulation.*

Increasing the number of the electric vehicles in the simulation makes the demand spike higher. This is the observation that motivated me to design the prediction and allocation messages that were presented in Section 4.5.

5.2 Raspberry Pi Demonstration

This section provides the details of the Raspberry Pi demonstration discussed in Section 4.6.

5.2.1 Hardware

The demonstration uses Raspberry Pi model B computers. These are small, low powered computers that only have 512 MB of RAM and a 700MHz processor. The low performance of the Raspberry Pi accurately represents the embedded computers on smart devices that PowerMatcher might run on.

An existing configuration of Raspberry Pi was made available for this project. A relay board is connected to the GPIO pins of the Raspberry Pi. A 12V DC bus powered by the mains is connected to one of the relays. Another relay controls a 20W bulb. Using the GPIO pins to close the relays, the bulb can be switched on and powered by the 12V DC supply.

In this demonstration, the light bulb was used to represent an electric vehicle.

5.2.2 Scenario

The demonstration consists of two agents, each running on separate Raspberry Pi computers. An Auctioneer agent ran on one device, while the electric vehicle device agent and simulation ran on the other. The auctioneer agent creates a WebSocket server which the electric vehicle connects to using a WebSocket client, as described in Section 3.4. The client and server are communicating over the local WiFi network. A diagram of this scenario is shown in Figure 5.5.

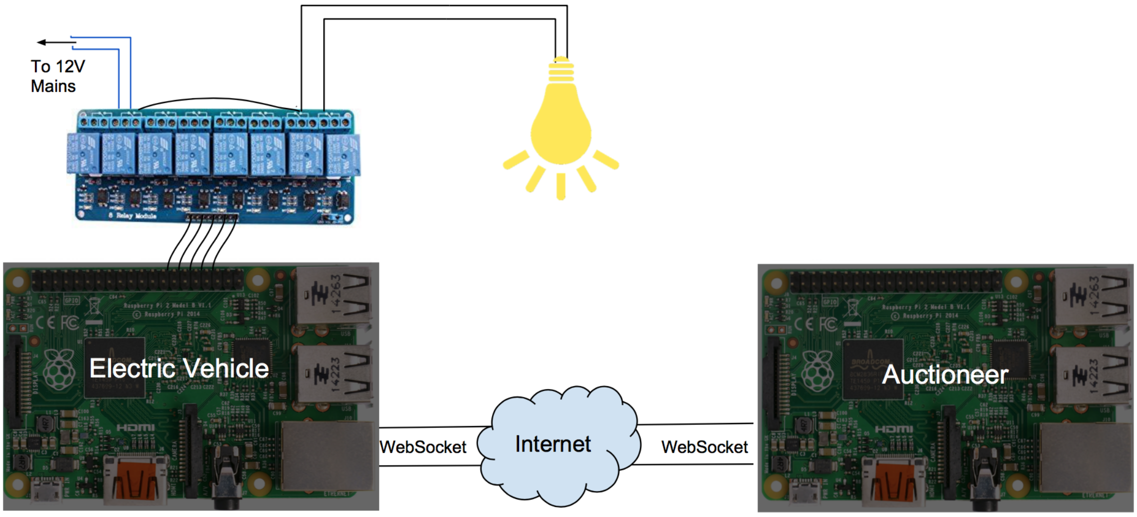


Figure 5.5: *The configuration for the Raspberry Pi demonstration.*

5.2.3 Outcome

When running, the electric vehicle agent successfully sends bids to the auctioneer agent. The electric vehicle follows the same behaviour as the simulation shown in Figure 5.2. When the device agent sets the car to be charging, it also successfully switches on the light bulb connected to the relay.

This simple scenario demonstrates that PowerMatcher can be configured and run on small embedded devices. The physical switching on and off of the light shows that PowerMatcher can be easily used to control real loads. Further details of challenges faced while implementing both the simulations and the demonstration are discussed below.

5.3 Open Service Gateway initiative (OSGi)

PowerMatcher was built using the OSGi framework and as a result, becoming familiar with OSGI was an essential component of this project [65]. An overview of the benefits and functionality of OSGi are given below. Following this, issues that were caused by OSGi during development time are discussed.

5.3.1 Overview of OSGi

OSGi is a modular framework for Java. In OSGi, a system is made up of bundles. A bundle is a standard Java Archive (Jar) file that contains extra manifest files describing services that the bundle can export. The bundles dependencies are managed by the OSGi framework, several benefits of which are highlighted by the OSGi Alliance [66]. The most notable of these is the reduced complexity associated with managing large distributed systems as a result of being modular. Each bundle hides its internal working, only exposing necessary services. This allows the bundle to change its internal workings without disrupting any other bundle that may depend on it.

The OSGi framework must run inside a container. A beneficial feature is that any bundle can run in any container. The PowerMatcher developers recommend using the Apache Felix container [67].

OSGi also provides a dynamic life cycle. Bundles are installed in the running container when first deployed. They can then be stopped and started throughout the lifetime of the application. Stopping one bundle does not stop the whole system, as other bundles can continue functioning even if they depend on that particular bundle.

As part of the dynamic life cycle, OSGi also provides versioning. A bundle repository can be created and as bundles are changed, new versions are published to the repository. If a bundle depends on a certain version of a different bundle then the OSGi framework can install and start the old version. This allows developers to safely move onto successor versions without stopping the system.

In PowerMatcher, each device agent is a bundle. Modularity is useful in PowerMatcher, as the whole system may not be completely under one persons control. If one device agent decides to leave the PowerMatcher system, it will not break the whole system.

5.3.2 Bnd and Bndtools

Bnd is a tool that allows for bundles to be automatically built in the OSGi framework. It reduces a lot of the repetition involved in creating manifest files while also providing tools to package bundles into Jar files.

Bndtools is a plugin for the Eclipse IDE that provides a graphic interface for bnd. Bndtools makes use of the OSGi lifecycle to provide continuous building. As bundles are developed, they are redeployed into the framework. Bndtools can also host repositories for bundles.

5.3.3 Problems Faced with OSGi

OSGi provides several benefits when used by a large organisation with developers who are experienced with the framework. However, OSGi has a very steep learning curve that makes using the framework hard for the individual developer. Documentation is spread across Bnd [68], OSGi Enroute [69] and Apache Felix [70]. The tutorials provided by these organisations give very simple, specific introductions to features, then skip to complicated documentation for advanced features. This leaves navigating an existing OSGi project quite difficult for a developer who is not familiar with the technology.

An issue I faced while developing in PowerMatcher was managing versions. Editing existing bundles does not save them to the bundle repository, meaning that when a bundle is utilised, the old version is used, ignoring new edits. For this project, a solution was eventually found that forced a bundle to get its dependencies directly from their file rather than the repository. In future development, this would need to change to make use of the versioning feature provided by OSGi.

5.4 PowerMatcher on the Raspberry Pi

The original design had the Raspberry Pi at the centre of the project. However, due to limitations discussed below, it could only be used as a side to the main simulation as a proof of concept.

5.4.1 Remote Configuration

As mentioned previously, the PowerMatcher developers recommend using the Apache Felix OSGi container as well as the Apache Felix Web Console to configure bundles. This is a web console that provides a graphical web page alternative to using the command line. In a real world scenario, it would be too time-consuming to connect a monitor to each Raspberry Pi and configure the bundles required for a device agent. In an ideal scenario, the bundles would be packaged correctly so they are ready to be deployed on the Raspberry Pi. However, this option could not be easily found. Consequently, for this project, the Apache Felix Web Console was configured to allow remote access. This allowed PowerMatcher to be deployed on several devices and configured remotely.

5.4.2 Controlling the Relay

In order to follow OSGi standards, I intended to build a bundle that would act as an application programming interface (API) for the relay module connected to the Raspberry Pi. The relay module is configured using the GPIO pins. With help from the developer, I attempted to utilise a third-party library to control the GPIO pins [71]. However, due to dependency issues within the library, this method could not be used. This hindered development time and resulted in the Raspberry Pi element of the project being put on hold.

For the purpose of demonstration, a small PowerMatcher system was set up between two Raspberry Pi computers as described in Section 5.2. Writing to a configuration file for the GPIO pin allowed manipulation of the relay to control a light. This method provided a good visual demonstration of PowerMatcher but was not reliable and was discovered too late to allow in depth usage of the hardware.

5.5 Simulation Environment

All PowerMatcher agents rely on a runtime environment to schedule the bidding process. I extended PowerMatcher to use a simulation runtime environment, allowing scenarios to be simulated in PowerMatcher at different rates.

5.6 Logging and Analysis

PowerMatcher does not provide much useful telemetry. As I could not rely on the feedback from hardware, logging was needed to track the performance of the system in each scenario.

5.6.1 CSV Logging

PowerMatcher does include an OSGi bundle that creates comma separated value (CSV) logs for agent bids and prices. This bundle uses the observer pattern. Device agents publish bid and price events to the CSV observer, which then formats and writes them to a log file.

To get more information from the system, I implemented event types for an electric vehicle, solar panel, wind turbine and battery. This enabled me to collect the electricity usage and production of each device to create detailed logs of simulations.

5.6.2 Aggregation and Graphing

To visualise the data collected, I wrote Python scripts to aggregate and plot it. Plot.ly provides an API for online visualisation and analysis of data [72].

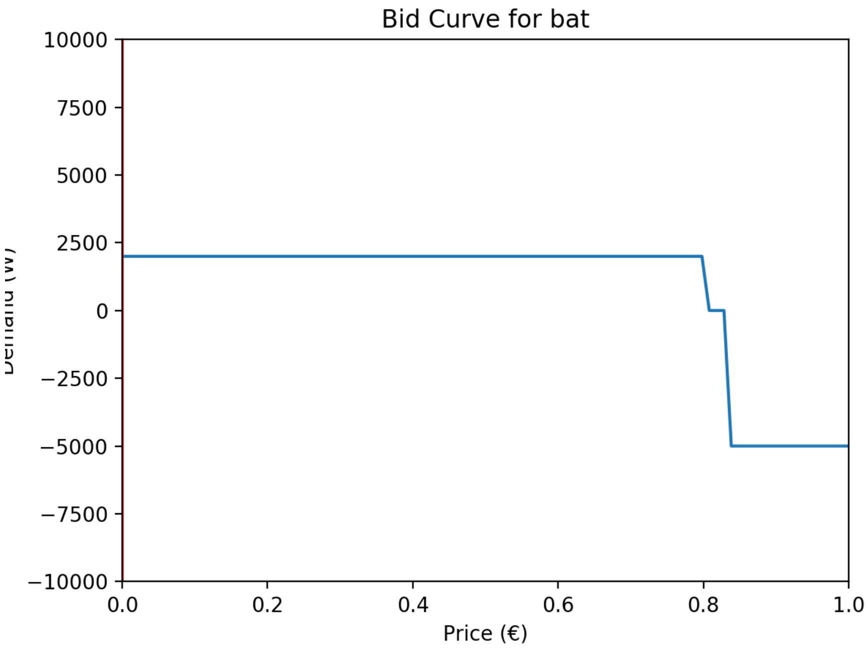


Figure 5.6: *An example of the live bid visualisation for a battery Device agent.*

Console logs are too congested to track multiple agents behaviours during a simulation. To give more insight during a simulation, I created a Python script that would create live bid graphs for each agent using the Matplotlib library [73]. An example of this visualisation is shown in Figure 5.6.

Chapter 6

Discussion and Conclusion

The electricity grid will see a radical change in the coming years. As residential renewable energy sources become more popular, the task of coordinating the electricity grid becomes more difficult. Balancing supply and demand is currently a centralised task performed by the grid operator and assisted by demand response operators. However as the electricity grid becomes decentralised, the task of balancing supply and demand must do the same. The new decentralised methods will depend on smart grid applications to control devices.

This dissertation presents PowerMatcher as a tool to coordinate the electricity grid. PowerMatcher has a large learning curve which is a barrier to its use in small scale projects such as this. The learning curve is due to limited documentation and use of niche technologies. This project has laid a foundation for future exploration of PowerMatcher's capabilities. Future projects that could be built upon this are outlined below in Section 6.1.

During the initial work of simulating the scenario discussed in Chapter 4, I noticed the need for an extension to PowerMatcher. This extension provides a prediction mechanism. I have provided a prototype design for the messages needed for this communication in Section 4.5. These messages have not been fully developed, however I have implemented the basic message containers needed for these messages. In future work, these message containers could be built upon to fully implement the prediction mechanism.

6.1 Future Work

PowerMatcher has the potential to be used in a variety of methods to solve problems faced by the grid. This section outlines some of these applications.

6.1.1 Real Prices

In this project, the price put on electricity is arbitrary. Each Device agent bid is constructed by calculating a percentage of desire for electricity which is applied to the maximum price as set by the Auctioneer. To further integrate PowerMatcher with the grid, the Auctioneer could communicate the real price of electricity, as determined by the current wholesale market. Users could then set maximum prices they are willing to pay for electricity. Devices would then construct bids based on these new prices. Using real time pricing would allow PowerMatcher to perform automated real time pricing DR as discussed in Section 2.4.2. Real time prices for the Irish market are available from the Single Electricity Market Operator (SEMO) website [74].

A challenge with devices having a maximum price that they will pay for electricity is that if the real price of electricity is too high then devices may never turn on. Bids would have to be constructed to take this into account.

6.1.2 Curtailment

For economic and environmental reasons, maximising the electricity produced by renewable sources is beneficial. However, renewable energy is intermittent and in most cases dependant on the weather. This introduces a security risk: if all electricity in a country is being generated from wind turbines, when the wind unexpectedly drops there may not be a way to quickly produce electricity from other sources. For this reason, grid operators must set a limit on how much generation can come from renewable sources. In Ireland, this limit is quite high at 55%, with trials of 60% [75]. This means that at any moment, renewable sources can make up to 55% of the current generation. If there is low demand, or high winds, there may be enough electricity available that more than 55% of the generation could come from renewable sources. In this case, the energy must be curtailed, meaning that wind turbines must be turned off or turned down. Eirgrid reports that 177GWh of potential wind energy was curtailed in 2016 [75].

Eirgrid have aimed to increase the penetration of renewable energy to 70% by 2020 through the D3 Programme [76]. PowerMatcher could be used to achieve this by taking current wind production into account. This project had a goal of using the main electricity grid as little as possible. Adding a device agent that represents wind production on the electricity grid to the system could allow the electricity grid to be part of the system, rather than an external component. The system could increase demand by charging batteries and electric vehicles to reduce the amount of wind energy curtailed.

6.1.3 Real Battery, Solar, and Weather Information

This Raspberry Pi demonstration for this project took place in a lab that also contains a battery that is charged using a solar PV panel which is installed on the roof. A weather station is also installed to give accurate solar measurements. As these are already installed, the demonstration for this project could be extended to make use of them. Data from the weather station could be used to develop an agent for the solar panel. When there is a lot of solar energy available, the PowerMatcher system could use electricity from the battery rather than the main electricity grid.

6.1.4 Nanogrid Control

The majority of smart grid applications build upon the existing grid structure. Nanogrids are a concept that aim to build the smart grid from the bottom up [7]. There is often an analogy made between the structure of the internet and the grid structure if nanogrids were in place. ‘A nanogrid is a single domain for voltage, reliability, and administration’ [77]. A nanogrid consists of one controller that has one to many loads connected to it, including a source of electricity. It also has a gateway through which it can connect to other nanogrids. Communication between grids allows the connection of many nanogrids to make up one large grid [78]. Figure 6.1 shows a diagram of a nanogrid.

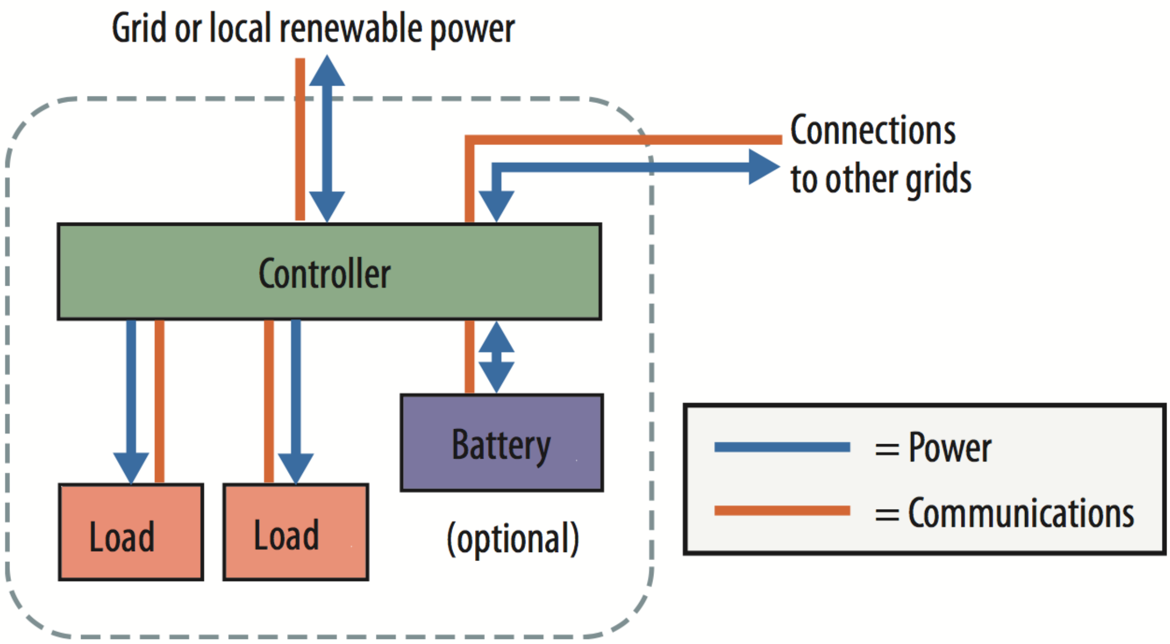


Figure 6.1: *Conceptual diagram of a nanogrid [7]*

It would be interesting to adapt PowerMatcher to control a nanogrid. An Auctioneer agent could serve as the main controller for the nanogrid with many device agents below it. A modification is needed to allow two Auctioneer agents controlling different nanogrids to communicate. A proxy could be added so that each Auctioneer agent can appear as a device agent to the other. This would allow the nanogrids to make bids to, and receive price from other nanogrids.

6.2 Learnings

This section gives a brief overview of what I have learned during the course of this research.

6.2.1 Smart Grids

This project allowed me to gain an in-depth knowledge of the evolving grid. Having an interest in sustainability and a general understanding of how the grid worked, a smart grid project appealed to me. I have become very familiar with the issues around balancing supply and demand, and the current methods used to solve these. There is a huge amount to learn in this area. My main focus was on transactive energy techniques and other solutions to the decentralised grid problem.

6.2.2 Hardware

As I have never used a Raspberry Pi, or done much work with hardware, this project gave me an opportunity to expand my goals past software and into real world devices.

6.2.3 OSGi

As mentioned in Section 5.3, a huge barrier for this project was the OSGi framework. Starting as a basic Java developer, I have learned a lot about the language and the OSGi framework. OSGi is a powerful tool in reducing the complexity of modularity and distributed applications. It was challenging to learn a framework while simultaneously developing a project using it but was the best way to understand the inner workings of it.

6.2.4 PowerMatcher

Having little knowledge of smart grid applications, I had never heard of PowerMatcher before my research. Dedicating a full year of work to one tool has allowed me to gain an in-depth understanding of the theories behind it and the functionality of the tool itself.

Bibliography

[1] “Final Report on the August 14, 2003 Blackout in the United States and Canada: Causes and Recommendataions,” U.S - Canada Poweer System Outage Task Force, Tech. Rep., Jan. 2004. [Online]. Available: https://www.ferc.gov/industries/electric/indus-act/reliability/blackout/ch1-3.pdf

[2] “Dispelling the nuclear baseload myth,” Mar. 2016. [Online]. Available: http://energypost.eu/dispelling-nuclear-baseload-myth-nothing-renewables-cant-better/

[3] “PowerMatcher under the Hood.” [Online]. Available: http://flexiblepower.github.io/technology/powermatcher/

[4] “Device Agent Bids | PowerMatcher.” [Online]. Available: https://fan-ci.sensorlab.tno.nl/builds/powermatcher-documentation/master/html/Bids.html

[5] “Aggregation | PowerMatcher.” [Online]. Available: https://fan-ci.sensorlab.tno.nl/builds/powermatcher-documentation/master/html/Aggregation.html

[6] “Equilibrium Price | PowerMatcher.” [Online]. Available: https://fan-ci.sensorlab.tno.nl/builds/powermatcher-documentation/master/html/Equilibrium.html

[7] B. Nordman, “Think Globally, Distribute Power Locally: The Promise of Nanogrids,” Sep. 2012.

[8] G. L. Barbose and N. R. Darghouth, “Tracking the Sun IX: The Installed Price of Residential and Non-Residential Photovoltaic Systems in the United States,” Lawrence

Berkeley National Laboratory, Tech. Rep. LBNL-1006036, Aug. 2016. [Online]. Available: https://emp.lbl.gov/sites/default/files/tracking_the_sun_ix_report.pdf

[9] "powermatcher: PowerMatcher - The Java implementation of the PowerMatcher, including the API, the core, a couple of examples, a remote implementation using websockets and a visualisation of the con.." Apr. 2017. [Online]. Available: https://github.com/flexiblepower/powermatcher

[10] JS, "How electricity grew up? A brief history of the electrical grid," Oct. 2012. [Online]. Available: https://power2switch.com/blog/how-electricity-grew-up-a-brief-history-of-the-electrical-grid/

[11] A. Ipakchi and F. Albuyeh, "Grid of the future," *IEEE Power and Energy Magazine*, vol. 7, no. 2, pp. 52–62, Mar. 2009.

[12] M. Dreidy, H. Mokhlis, and S. Mekhilef, "Inertia response and frequency control techniques for renewable energy sources: A review," *Renewable and Sustainable Energy Reviews*, vol. 69, pp. 144–155, Mar. 2017. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1364032116309212

[13] E. Burleson, "Paris Agreement and Consensus to Address Climate Challenge," *ASIL INSIGHT, Forthcoming*, Jan. 2016. [Online]. Available: https://ssrn.com/abstract=2710076

[14] N. Panwar, S. Kaushik, and S. Kothari, "Role of renewable energy sources in environmental protection: A review," *Renewable and Sustainable Energy Reviews*, vol. 15, no. 3, pp. 1513–1524, Apr. 2011. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1364032110004065

[15] C. Day and G. Day, "Climate change, fossil fuel prices and depletion: The rationale for a falling export tax," *Economic Modelling*, vol. 63, pp. 153–160, Jun. 2017. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0264999317300731

[16] V. Court and F. Fizaine, “Long-Term Estimates of the Energy-Return-on-Investment (EROI) of Coal, Oil, and Gas Global Productions,” *Ecological Economics*, vol. 138, pp. 145–159, Aug. 2017. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0921800915303815

[17] “A report from the Commission to the European Parliament, the Council, the European Economic and Social Committee and the Commitee of the Regions: Renewable Energy Progress Report,” European Commision, Tech. Rep., Jan. 2017. [Online]. Available: http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:52017DC0057

[18] B. Ang and B. Su, “Carbon emission intensity in electricity production: A global analysis,” *Energy Policy*, vol. 94, pp. 56–63, Jul. 2016. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0301421516301458

[19] K. Kok, B. Roossien, P. MacDougall, O. v. Pruissen, G. Venekamp, R. Kamphuis, J. Laarakkers, and C. Warmer, “Dynamic pricing by scalable energy management systems - Field experiences and simulation results using PowerMatcher,” in *2012 IEEE Power and Energy Society General Meeting*, Jul. 2012, pp. 1–8.

[20] C. E. Ministerial, “Electric Vehicles Initiative Government Fleet Decleration,” 2016. [Online]. Available: https://www.iea.org/media/topics/transport/EVI_Government_Fleet_Declaration.pdf

[21] “India’s Big Push To Go All-In On Electric Vehicles By 2030,” May 2017. [Online]. Available: http://www.nasdaq.com/article/indias-big-push-to-go-allin-on-electric-vehicles-by-2030-cm786473

[22] F. Lambert, “All new cars mandated to be electric in Germany by 2030 [Updated],” Jun. 2016. [Online]. Available: https://electrek.co/2016/06/14/all-new-cars-mandated-electric-germany-2030/

[23] Q. Qdr, "Benefits of demand response in electricity markets and recommendations for achieving them," *US department of energy*, 2006. [Online]. Available: https://stuff.mit.edu/afs/athena.mit.edu/dept/cron/Backup/project/urban-sustainability/Old%20files%20from%20summer%202009/Ingrid/Urban%20Sustainability%20Initiative.Data/PDF/doe_demand_response_rpt_t---ss_feb_17_06_final-1713783046/doe_demand_response_rpt_t---ss_feb_17_06_final.pdf

[24] N. Good, K. A. Ellis, and P. Mancarella, "Review and classification of barriers and enablers of demand response in the smart grid," *Renewable and Sustainable Energy Reviews*, vol. 72, pp. 57–72, May 2017. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1364032117300436

[25] J. S. Vardakas, N. Zorba, and C. V. Verikoukis, "A Survey on Demand Response Programs in Smart Grids: Pricing Methods and Optimization Algorithms," *IEEE Communications Surveys Tutorials*, vol. 17, no. 1, pp. 152–178, 2015.

[26] E. A. M. Klaassen, "Demand response benefits from a power system perspective," 2016. [Online]. Available: https://pure.tue.nl/ws/files/43797372/20161122_Klaassen.pdf

[27] H. T. Haider, O. H. See, and W. Elmenreich, "A review of residential demand response of smart grid," *Renewable and Sustainable Energy Reviews*, vol. 59, pp. 166–178, Jun. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1364032116000447

[28] "Smart Grid Dashboard - Market Price." [Online]. Available: http://smartgriddashboard.eirgrid.com/

[29] "All you need to know about NightSaver meters? - Electric Ireland Blog," 2014. [Online]. Available: https://www.electricireland.ie/news-media/article/news/2014/05/21/all-you-need-to-know-about-the-nightsaver-meter

[30] "OpenADR Alliance." [Online]. Available: http://www.openadr.org/

[31] T. G. A. Council, "GridWise Transactive Energy Framework version 1.0," Jan. 2015. [Online]. Available: http://www.gridwiseac.org/pdfs/te_framework_report_pnnl-22946.pdf

[32] S. Chen and C.-C. Liu, "From demand response to transactive energy: state of the art," *Journal of Modern Power Systems and Clean Energy*, vol. 5, no. 1, pp. 10–19, Jan. 2017. [Online]. Available: http://link.springer.com/10.1007/s40565-016-0256-x

[33] D. J. Hammerstrom, S. E. Widergren, and C. Irwin, "Evaluating Transactive Systems: Historical and Current U.S. DOE Research and Development Activities," *IEEE Electrification Magazine*, vol. 4, no. 4, pp. 30–36, Dec. 2016.

[34] K. Kok and S. Widergren, "A Society of Devices: Integrating Intelligent Distributed Resources with Transactive Energy," *IEEE Power and Energy Magazine*, vol. 14, no. 3, pp. 34–45, May 2016.

[35] "Pacific Northwest Smart Grid Demonstration Project." [Online]. Available: http://www.pnwsmartgrid.org/about.asp

[36] "Transactive Coordination Signals," The Pacific Northwest Smart Grid Demonstration Project, Tech. Rep. PNWD-4402, Aug. 2013. [Online]. Available: http://www.pnwsmartgrid.org/docs/Transactive_Coordination_Signals.pdf

[37] "VOLTTRON | Department of Energy." [Online]. Available: https://energy.gov/eere/buildings/volttron

[38] S. Katipamula, J. Haack, G. Hernandez, B. Akyol, and J. Hagerman, "VOLTTRON: An Open-Source Software Platform of the Future," *IEEE Electrification Magazine*, vol. 4, no. 4, pp. 15–22, Dec. 2016.

[39] "PNNL: : Rewards for Responsible, Real-time Energy Management." [Online]. Available: http://energyenvironment.pnnl.gov/highlights/highlight.asp?sector=53&id=1588

[40] J. Horta, D. Kofman, and D. Menga, “Novel paradigms for advanced distribution grid energy management,” 2016. [Online]. Available: http://www.the-blockchain.com/docs/Novel%20paradigms%20for%20advanced%20distribution%20grid%20energy%20management.pdf

[41] J. Mattila, T. Seppl, C. Naucler, R. Stahl, M. Tikkanen, A. B\a adenlid, and J. Seppl, “ETLA Working Papers,” 2016. [Online]. Available: http://www.etla.fi/wp-content/uploads/ETLA-Working-Papers-43.pdf

[42] “Blockchain-based microgrid gives power to consumers in New York.” [Online]. Available: https://www.newscientist.com/article/2079334-blockchain-based-microgrid-gives-power-to-consumers-in-new-york/

[43] “TransActive Grid | Secure, transactional control of utility systems.” [Online]. Available: http://transactivegrid.net/

[44] D. T. Ton and M. A. Smith, “The U.S. Department of Energy’s Microgrid Initiative,” *The Electricity Journal*, vol. 25, no. 8, pp. 84–94, Oct. 2012. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1040619012002254

[45] P. Borazjani, N. I. A. Wahab, H. B. Hizam, and A. B. C. Soh, “A review on microgrid control techniques,” in *2014 IEEE Innovative Smart Grid Technologies - Asia (ISGT ASIA)*, May 2014, pp. 749–753.

[46] M. Khodayar, S. Manshadi, and A. Vafamehr, “The short-term operation of microgrids in a transactive energy architecture,” *The Electricity Journal*, vol. 29, no. 10, pp. 41–48, Dec. 2016. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1040619016301695

[47] K. Kok, “The PowerMatcher: Smart Coordination for the Smart Electricity Grid,” Ph.D. dissertation, Vrije Universiteit, Amsterdam, 2013. [Online]. Available: http://hdl.handle.net/1871/43567

[48] "PowerMatcher under the Hood - Objective Agent," May 2017. [Online]. Available: http://flexiblepower.github.io/technology/powermatcher/

[49] B. Roossien, "Field-test upscaling of multi-agent coordination in the electricity grid," in *20th International Conference and Exhibition on Electricity Distribution - Part 1, 2009. CIRED 2009*, Jun. 2009, pp. 1–4.

[50] M. Wooldridge, *An Introduction to MultiAgent Systems.* John Wiley & Sons, Jun. 2009.

[51] "Architecture | PowerMatcher," May 2017. [Online]. Available: https://fan-ci.sensorlab.tno.nl/builds/powermatcher-documentation/master/html/Architecture-PowerMatcher.html

[52] I. Fette, "The WebSocket Protocol." [Online]. Available: https://tools.ietf.org/html/rfc6455

[53] M. Hoefling, F. Heimgaertner, M. Menth, and H. Bontius, "Traffic estimation of the PowerMatcher application for demand supply matching in smart grids," in *2015 International Conference and Workshops on Networked Systems (NetSys)*, Mar. 2015, pp. 1–6.

[54] F. Bliek, A. v. d. Noort, B. Roossien, R. Kamphuis, J. d. Wit, J. v. d. Velde, and M. Eijgelaar, "PowerMatching City, a living lab smart grid demonstration," in *2010 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT Europe)*, Oct. 2010, pp. 1–8.

[55] "PowerMatching City | Home," Jun. 2017. [Online]. Available: http://www.powermatchingcity.nl/site/pagina.php?

[56] "Smart Home - House for smart people," Jun. 2017. [Online]. Available: http://www.smarthouse-smartgrid.eu/

[57] P. Kempker, N. v. Dijk, W. Scheinhardt, H. v. d. Berg, and J. Hurink, "Optimization of charging strategies for electric vehicles in PowerMatcher-driven smart energy grids," in *Proceedings of the 9th EAI International Conference on Performance Evaluation Methodologies and Tools.* ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2016, pp. 242–249. [Online]. Available: http://dl.acm.org/citation.cfm?id=2897439

[58] "Tesla Solar," Jun. 2017. [Online]. Available: https://www.tesla.com/solar

[59] S. Shao, M. Pipattanasomporn, and S. Rahman, "Grid Integration of Electric Vehicles and Demand Response With Customer Choice," *IEEE Transactions on Smart Grid*, vol. 3, no. 1, pp. 543–550, Mar. 2012.

[60] S. Pfenninger and I. Staffell, "Renewables.ninja." [Online]. Available: https://www.renewables.ninja/

[61] ——, "Long-term patterns of European PV output using 30 years of validated hourly reanalysis and satellite data," *Energy*, vol. 114, pp. 1251–1265, Nov. 2016. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0360544216311744

[62] "Tesla Powerwall." [Online]. Available: https://www.tesla.com/powerwall#technical-specs

[63] "EF-Pi Flexible Energy," Aug. 2017. [Online]. Available: http://flexible-energy.eu/ef-pi/

[64] M. Yilmaz and P. T. Krein, "Review of the Impact of Vehicle-to-Grid Technologies on Distribution Systems and Utility Interfaces," *IEEE Transactions on Power Electronics*, vol. 28, no. 12, pp. 5673–5689, Dec. 2013.

[65] "OSGi Alliance The Dynamic Module System for Java." [Online]. Available: https://www.osgi.org/

[66] "Benefits of Using OSGi OSGi Alliance," 2017. [Online]. Available: https://www.osgi.org/developer/benefits-of-using-osgi/

[67] "Apache Felix - Documentation." [Online]. Available: https://felix.apache.org/documentation.html

[68] "BND Documentation Introduction." [Online]. Available: http://bnd.bndtools.org/chapters/110-introduction.html

[69] "Enroute OSGi Documentation." [Online]. Available: http://enroute.osgi.org/book/210-doc.html

[70] "Apache Felix - Welcome to Apache Felix." [Online]. Available: https://felix.apache.org/

[71] "Tutorial: Raspberry Pi GPIO with OSGi Services - Eclipsepedia." [Online]. Available: https://wiki.eclipse.org/Tutorial:_Raspberry_Pi_GPIO_with_OSGi_Services

[72] "Visualize Data, Together." [Online]. Available: https://plot.ly/

[73] "Matplotlib: Python plotting Matplotlib 2.0.1 documentation," Tech. Rep. [Online]. Available: https://matplotlib.org/

[74] SEMO, "SEMO - Single Electricity Market Operator," Dec. 2009. [Online]. Available: http://www.sem-o.com/Pages/default.aspx

[75] "Annual-Renewable-Constraint-and-Curtailment-Report-2016-v1.0.pdf," Eirgrid and SONI, Tech. Rep., Apr. 2017. [Online]. Available: http://www.eirgridgroup.com/site-files/library/EirGrid/Annual-Renewable-Constraint-and-Curtailment-Report-2016-v1.0.pdf

[76] Eirgrid, "DS3 Programme," 2017. [Online]. Available: http://www.eirgridgroup.com/how-the-grid-works/ds3-programme/#comp_000056cb5b8e_00000006da_78f0

[77] B. Nordman, “Evolving our electricity systems from the bottom up.” [Online]. Available: https://eta-intranet.lbl.gov/sites/default/files/nano.pdf

[78] B. Nordman and K. Christensen, “The need for communications to enable DC power to be successful,” in *2015 IEEE First International Conference on DC Microgrids (ICDCM)*, Jun. 2015, pp. 108–112.