Information and Communication Infrastructure for Future Power Grids

Final ECN report on theme C of the project Electricity Infrastructure of the Future (EIT)

Koen Kok

Bart Roossien

Gerben Venekamp

Josco Kester

ECN-E--11-032 OCTOBER 2011

Contents

1. Introduction

1.1 Research project Electrical Infrastructure of the future (EIT)

This report describes the results of the ECN contribution to the topic *Information and communication (infra)structure* (theme C) of the project Energy Infrastructure of the Future (EIT-Elektriciteitsinfrastructuur van de Toekomst). The EIT-project is a joint effort of TU/e, ECN and KEMA, coordinated by TU/e. The project aims at researching the following themes:

A. Functional specifications and design of *efficient and flexible transport systems*
B. Design and performance of a distribution grid *fully controlled by power electronics*
C. Specification and layout of *information and communication (infra)structure* needed for a reliable and sustainable energy system.
D. Opportunities for the application, assessment of economical aspects and *offering market perspectives*.

Research at the TU/e on theme C is still continuing and will be reported later in a separate report. The results of theme A, B, and D have been or will be reported in separate reports as well (by TU/e, TU/e and KEMA respectively).

The rationale for the EIT-project is that requirements for the electrical infrastructure of the future will be fundamentally different from the current requirements. The current infrastructure can't satisfy those requirements. It is too passive, not intelligent enough and not capable to control the changing conditions – which makes it vulnerable. The following trends are emerging:

- The generation of electrical energy will change structurally: much local generation, stochastic output and need for storage.
- Growing energy needs, need for energy management and system integration within the boundary conditions of the primary process of the end user, the desire for energy savings and demand response.
- Changing demands of customers and needs of the society as for quality and reliability, increasing sensitivity of equipment and industrial processes for tolerances in the voltage supplied.
- Individualisation of services to the customer, premium power for privileged applications, market oriented solutions for control of bottlenecks in the system and combination of services.

The resulting problems for the grids will need to be solved by a strategy aiming at:

- *Efficient and flexible transport systems*, with implementations that are acceptable to society, have low losses, high capacity and a high availability through in-built security and recovery, combination of services
- *Application of new technologies on the basis of power electronics*, such as DC interfaces, power flow controller and short circuit current limiters; new concepts for operation, coordination and management
- *Use of the opportunities offered by Information and Communication Technologies*, such as enhanced automation, intelligent data collection and processing, local coordination and control
- *Offering of market perspectives* for new technologies and concepts; transition management.

The authors wish to thank Agentschap NL for partial funding of this research under the EOS-LT program for long term energy research of the Dutch ministry of Economic Affairs and Innovation.

1.2 Sustainable Electricity Sources

In electricity generation two inter-related movements can be seen, both of paramount importance for the way the electricity system will be managed in the future:

1. The increase of electricity generated from **sustainable energy sources**.

2. **Decentralization** of electricity generation: electricity generating units are growing in numbers and moving closer to the load centers.

In this section and the next one we will describe these changes in more detail, and in the last section of this chapter we will describe the impact on management of the electricity system. Worldwide, two thirds of the electricity is still produced from fossil fuels (natural gas, oil and coal) while approximately 15% originates from nuclear sources [9]. Of the sustainable options for electricity generation, hydro energy is currently most significant in the worldwide power production (17%). Other sustainable energy sources (wind, solar, biomass, and geothermal) contribute for only about 2% to the worldwide electricity generation[1].
However, there are important drivers to reduce the fossil fuel dependency and to substitute fossil fuels for sustainable energy sources. Two important drivers behind this are:

- **Environmental concerns:** pollution and climate change. Most fossil fuels are used as input for a combustion process which emit pollutants such as aerosols (e.g., soot), sulfur oxides and nitrogen oxides. Further, fossil fuel usage is one of the greatest contributors to global warming due to greenhouse gas emissions.

- **Diversification of energy sources:** the energy need of most western economies is largely imported from outside those economies. As energy demand continues to grow, this external dependence could grow steeply in the next decades. Moreover, a substantial portion of fossil fuels are imported from politically unstable regions. A higher portion of sustainable energy in the energy mix reduces this dependency.

As said, hydro energy is the only sustainable energy source with a substantial share in today's electricity supply. Worldwide, approximately 17% of electricity is generated by hydro power generators. However, the growth potential for hydro power is limited. In many countries, the capacity increase is due to new small hydro power facilities, instead of large hydro power plants. These generators are connected to the medium voltage distribution grid.
With an annual growth of 25 to 30%, wind energy is becoming the second largest sustainable energy source for power generation. In 2008, the capacity installed worldwide was 121 GW [24] (3.2% of total power generation capacity). With an annual growth of 25%, the wind generation capacity in 2020 will be 1750 GW, i.e., a share of at least 25% of the worldwide power generation capacity. In 2008, Germany had 24 GW wind generation capacity installed with a production share of 7.5%. Among the countries with the largest wind generation capacity in 2008 are the USA (25 GW), Spain (17 GW) and China (17 GW). Initially, wind turbines with a capacity up to 1000 kW (solitaire or in a wind park) were connected to the distribution grid. Today, however, very large wind turbines with a generation up to 5 MW each are installed offshore in large wind parks. Since the total generation capacity of these wind parks is often more than 100 MW, they are connected to the transmission grid. At the same time

[1] Sustainable Electricity Sources are also refered to as Renewable Energy Sources (RES). In the remainder of this text we will use these terms interchangeably.

there is a trend towards smaller wind turbines, i.e., turbines with a capacity of less than 50 kW. These turbines are situated near dwellings and connected to the low voltage distribution grid.

The most abounded sustainable energy source worldwide is solar energy. Solar energy can be converted to electricity through a thermal route using a steam cycle, as in conventional power plants, and through photovoltaic (PV) cells. The thermal technique is used in large plants (some hundreds of MW), so called concentrated solar power. Panels with PV cells are used in urban areas, mounted to the roofs of buildings and dwellings, and connected to the low voltage distribution grid. The total installed capacity of PV worldwide in 2007 was 9100 MWpeak of which 40% in Germany [33] . If the average annual growth factor of about 30% continues, the installed total worldwide generation capacity in 2020 may become 275 GWpeak. Although this will be only a few percent of the total installed generation capacity worldwide, locally the share of electricity production from PV may be much larger.

Biomass (wood, organic waste, etc.) has been used for power generation on a limited scale for decades. There is a large growth potential for this sustainable energy source. Different kinds of biomass can be cofired in coal fired power plants (10 to 30%). Biomass can also be converted into electricity in dedicated biomass plants. The size of these plants is smaller than conventional power plants, i.e., up to a few hundred MW. Another form of bioenergy is biogas. Biogas, from waste water treatment or anaerobic digestion of manure, can be used as a fuel for gas engines producing electrical power. These units have a capacity of some MWs and are connected to the medium voltage distribution grid.

Other sustainable energy sources are geothermal, wave and tidal energy. These energy sources are only available in specific regions, where they may be of significant importance. Geothermal electricity generation in Iceland is an example of this.

1.3 Distributed Generation

Another ongoing change in the electricity sector is a decentralization of generation. A growing share of the generation capacity is located in the distribution part of the physical infrastructure. This trend breaks with the traditional central plant model for electricity generation and delivery. For this type of generation the term *distributed generation* (DG) is used: the production of electricity by units connected to the distribution network or to a customer site.
Thus, DG units supply their generated power to the distribution network either directly or indirectly via a customer's private network (i.e., the network on the end-customer's premises, behind the electricity meter). Consequently, the generation capacities of individual DG units are small as compared to central generation units which are directly connected to the trans mission network. On the other hand, their numbers are much higher than central generation and their growth is expected to continue [15].

Sustainable or renewable energy sources (RES) connected to the distribution grid fall under the definition of DG. However not all RES are DG, as large-scale renewables, e.g., off-shore wind electricity generation, are connected to the transmission network. The same holds for Combined Heat and Power production (CHP – or Cogeneration). A CHP unit is an installation for generating both electricity and useable heat simultaneously. Dependent of their size, CHP units are either connected to the distribution grid (and, thus, fall under the definition of DG) or to the transmission grid. Table 1.1 categorizes different forms of CHP and RES into either large-scale generation or distributed generation.

Table 1.1: Characterization of Distributed Generation (adapted from [8])

	Combined Heat and Power	Renewable Energy Sources
Large-scale Generation	- Large district heating* - Large industrial CHP	- Large hydro** - Off-shore wind - Co-firing biomass in coal power plants - Geothermal energy - Concentrated solar power
Distributed Generation	- Medium district heating - Medium industrial CHP - Utility building CHP - Micro CHP	- Medium and small hydro - On-shore wind - Tidal energy - Biomass and waste incineration - Biomass and waste gasification - PV solar energy

* *Typically* $> 50 MW_e$; ** *Typically* $> 10\ MW_e$

There are a number of drivers behind the growing penetration of DG (adapted and augmented from [10]):

- **Environmental concerns; Depletion of Oil Reserves; Diversification of energy sources.** All three as described in Section 1.1.

- **Deregulation of the electricity market.** As a result of the deregulation, the long-term prospects for large-scale investments in power ge neration have become less apparent. Therefore, a shift of interest of investors from large-scale power generation plants to medium and small-sized generation can be seen. Investments in DG are lower and typically have shorter payback periods than those of the more traditional central power plants. Capital exposure and risk is reduced and unnecessary capital expenditure can be avoided by matching capacity increase with local demand growth.

- **Energy autonomy.** A sufficient amount of producing capacity situated in a local electricity network opens the possibility of intentional islanding. Intentional islanding is the transition of a sub-network to stand-alone operation during abnormal conditions on the externally connected network, such as outages or instabilities, e.g., during a technical emergency. In this manner, autonomy can be achieved on different scales, from single buildings to wide-area subsystems.

- **Energy Efficiency (i).** In general, distributed generation reduces energy transmission losses. Estimates of power lost in long-range transmission and distribution systems of western economies are of the order of 7%. By producing electricity in the vicinity of a consumption area, transport losses are avoided. There is, however, a concern that in cases where the local production outgrows the local consumption the transmission losses start rising again. But in the greater part of the world's distribution network we are far from reaching that point.

- **Energy Efficiency (ii).** Heat production out of natural gas can reach higher efficiency rates by using combined heat-power generation (CHP) instead of traditional furnace burners. CHP is a growing category of distributed generation, especially in regions where natural gas is used for heating. In Northern Europe, for instance, CHP is already commonly used in heating of large buildings, green houses and residential areas. The use of micro-CHP for domestic heating in single dwellings is also expected to breakthrough in the coming few years.

1.4 Demand Response

Because of the introduction of new types of power demand with high simultaneous demand peaks, such as Plug in (Hybrid) Electric Vehicles there is a growing need for flexible electricity generation. At the same time the flexibility of generation is decreasing, because many forms of sustainable energy generation (such as wind or solar energy) depend upon the momentary weather flexibility in generation new suppliers of flexibility will be needed in the near future. Demand response is an important candidate to supply this additional flexibility.

Demand response is the ability of electricity consuming installations and appliances to alter their operations in response to signals from the energy markets or electricity network operators in (near-)real time. Demand response can be achieved through avoidance of electricity use and/or by shifting load to another time period. At present, *price elasticity* of electricity demand is very low in the electricity markets. This means that the quantity in demand stays constant with a changing price. Higher elasticity in electricity demand would lead to:

1. A lower electricity price (see Figure 1.1). During the California energy crisis, a demand reduction of 5% during the periods of the highest price peaks would have reduced these prices by 50% [16].

2. Direct reduction of energy usage in the case demand response is achieved by avoidance of electricity use.

3. Lower usage of conventional peak power plants, which are generally inefficient and environmental unfriendly. For a number of European countries, a concentrated demand response effort of 20 to 75 hours per year leads to a 5% peak load reduction [12].

4. Lower market power of producers. The number of market parties competing during peak load periods is generally low. This gives peak power producers high market power leading to price inflation. Price elasticity at the demand side will counteract this by increasing competitiveness.

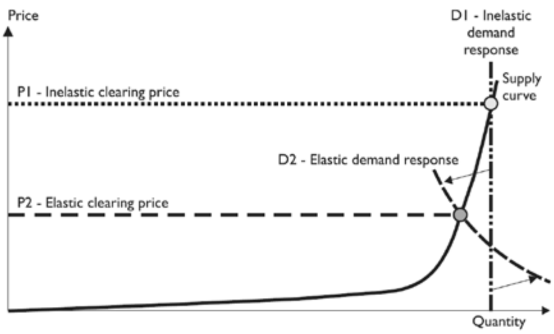

Figure 1.1: Impacts of Demand Elasticity on Wholesale Price [16].

Typical large flexible loads include different types of industrial processes, e.g., ground wood plants and mechanical pulping plants, electrolysis, arc furnaces, rolling mills, grinding plants, extruders, gas compressors, etc. In the commercial and residential sectors, the largest electrical loads can be made responsive: space heating, space cooling, tap water heating, refrigeration, freezing, washing or drying. Figure 1.2 gives average appliance load profiles for a generic European home. For all listed appliances, operation can be shifted in time except for the water heater (when it is a water kettle rather than a hot tap water vessel) and the oven/stove.

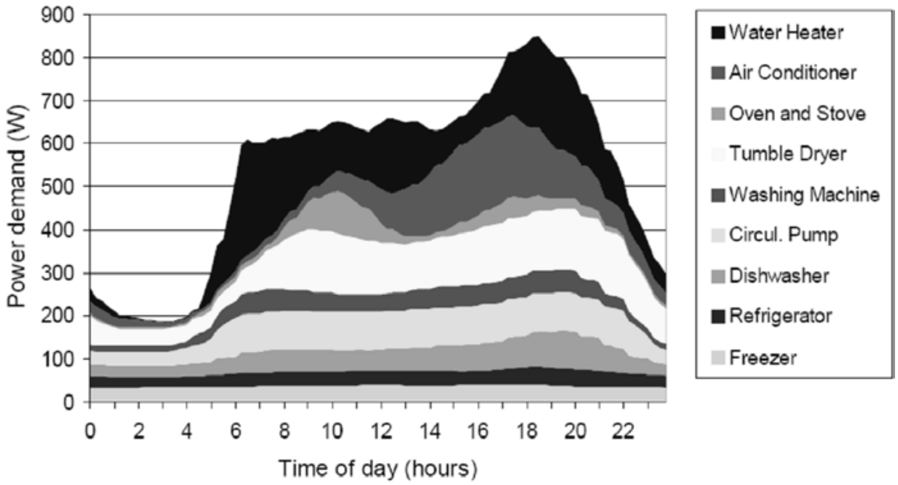

Figure 1.2: Appliance load profile of a generic European household averaged over a large number of households and over the period of one year. [29].

Household appliances can be involved in demand response in two ways: smart timing of appliance cycles and/or interruptions of appliance cycles. In smart cycle timing, the start of an appliance cycle is chosen such that the complete cycle lies in a preferable time period. For appliances such as washing machines and tumble dryers, this may involve a user action to indicate the preferred maximal ending time of the cycle. For a refrigerator or a freezer this means that the cycle starts before the maximum allowable temperature (or higher control temperature) is reached. In cycle interruption, the appliance cycle is interrupted for a certain period in time. For a washing machine or a tumble dryer, this means that during a running batch the heating process is interrupted for a certain time. For a refrigerator or a freezer this means that the cycle ends before the lower control temperature is reached.

Table 1.2: Demand response by household appliances: flexibility boundaries (adapted from [29])

Smart Timing of Appliance cycles	
Washing machine / dryer	Typical < 3 hrs; Maximum 9 hrs
Dishwasher	Typical < 6 hrs; Maximum > 12 hrs
Refrigerator / Freezer	Typical < 30 mins
Other appliances	Typical < 15 mins, . . . 1 hr

Interruptions of the Appliance cycle	
Washing machine	Typical < 10 mins
Dryer	Typical < 30 mins
Dishwasher	Typical < 10 mins
Refrigerator / Freezer	Typical < 15 mins
Other appliances	Typical < 15 mins

From the viewpoint of controllability, DG and DR are equivalent: increasing production has the same effect on the supply and demand balance as decreasing consumption, and vice versa. Due to this, demand response is sometimes treated as being a resource. As a result of the common nature of DG and DR (and distribution network connected electricity storage), the overarching term *Distributed Energy Resources* (DER) is used to refer to this threesome: DG, DR and storage.

1.5 Implications for Infrastructure Management

The decentralization of electricity generation is changing the characteristics of power generation in three aspects:

- **Intermittency:** The power production of most types of DG is intermittent in nature. Additionally, CHP units operated to follow heat demand are intermittent in nature as well. As stated before, with the growing share of these intermittent energy sources it becomes more difficult to follow the fluctuating electricity demand.
- **Cardinality:** As a result of generation decentralization, the number of electricity production units is growing rapidly while individual capacities are decreasing.
- **Location:** The location of power generation relative to the load centers is changing. Due to decentralization, the distance between generation units in the grid relative to the location of electricity consumption is becoming smaller. On the other hand, central renewable generation is moving further away from the load centers as large-scale wind farms are being built off-shore and large-scale solar power plants in desert areas.

Distributed generation does not fit into the standard paradigm of centralized control of a relatively small number big cental power plants. As distributed generation gradually levels with central generation, the centralized control paradigm will no longer suffice. The number of system components actively involved in the coordination task will be huge. Centralized control of such a complex system will reach the limits of scalability, computational complexity and communication overhead. The need to involve demand response in the coordination task only adds to this problem.

1.5.1 The Traditional Reaction: "Fit and Forget"

The traditional reaction to DG is *accommodation* in the existing electricity system, i.e., network and markets. This is the *"fit and forget"* approach. Distributed units are running free, beyond the control of the grid operator or the market-party to which the generated energy is delivered. The individual capacity of each separate DG unit is too small to be active on the wholesale market for electricity. Therefore, electricity supply companies treat DG as being negative demand: it is non-controllable and to a certain extend forecastable. As with renewable energy sources, a growth in DG decreases controllability and predictability in the electricity system. Again, the traditional reaction is to increase the capacity of regulating plants, while the total generation share of central generators goes down.

1.5.2 The Smart Reaction: Distributed Coordination

In the smart reaction, distributed generation, demand response, and future options for electricity storage, are integrated in the coordination mechanisms of the electricity system. As argued above, this can't be done by following the traditional paradigm of centralized control. Thus, a new paradigm for coordination tasks in electrical power systems. The new coordination mechanism is likely based on the state of the art in information and communication technology (ICT).

Before we look into the requirements of the needed ICT system, we take a closer look into the systems that need to play a role in the coordination task at hand. From the viewpoint of controllability, DG and DR are equivalent: increasing production has the same effect on the supply and demand balance as decreasing consumption, and vice versa. Due to this, demand response can be treated as a resource. The same holds for distribution network connected electricity storage. Due to this common nature, the overarching term Distributed Energy Resources (DER) is used to refer to this threesome: DG, DR and storage.

The high-level requirements of the coordination system that integrates DER in power systems operations and markets include:

- **Scalability**: A huge number of systems spread-out over a vast area will have to be involved in the coordination task. Especially on the level of the distribution grids, huge growth in the number of components actively involved in the coordination is expected. The coordination mechanism must be able to accommodate this growth.

- **Openness**: The information system architecture must be open: individual DER units can connect and disconnect at will and future types of DER —with own and specific operational characteristics— need to be able to connect without changing the implementation of the system as a whole. Therefore, communication between system parts must be uniform and stripped from all information specific to the local situation.

- **Multi-level Stakes**: The information system must facilitate a multi-actor interaction and balance the stakes on the global level (i.e., the aggregated behavior: reaction to energy market situation and/or network operator needs) and on the local level (i.e. DER operational goals).

- **Autonomy and Privacy**: In most cases, different system parts are owned or operated by different legal persons, so the coordination mechanism must be suitable to work over boundaries of ownership. Accordingly, the power to make decisions on local issues must stay with each individual local actor.

These requirements ask for a distributed system, also referred to as a multi-agent system (MAS) for a number of reasons:

- In multi-agent systems a large number of actors are able to interact, in competition or in cooperation. Local software agents focus on the interests of local sub-systems and influence the whole system via negotiations with other software agents. While the complexity of an individual agent can be low, the intelligence level of the global system is high.

- Multi-agent systems implement distributed decision-making systems in an open, flexible and extensible way. Communications between actors can be minimized to a generic and uniform information exchange.

- By combining multi-agent systems with micro-economic principles, coordination using economic parameters becomes possible. This opens the possibility for the distributed coordination process to exceed boundaries of ownership. The local agent can be adjusted by the local stakeholder, and does not fall under the rules and conditions of a central authority. Further, a Pareto efficient system emerges, i.e. a system that optimizes on a global level, while at the local level the interests of all individual actors are optimally balanced against each other.

2. Earlier Work: Market-based Balancing

2.1 Multi-agent Systems

The technology of multi-agent systems (MAS) provides a well-researched way of implementing complex distributed, scalable, and open ICT systems. A multi-agent system is a system of multiple interacting software agents. A software agent is a self-contained software program that acts as a representative of something or someone (e.g., a device or a user). A software agent is goal-oriented: it carries out a task, and embodies knowledge for this purpose. For this task, it uses information from and performs actions in its local environment or context. Further, it is able to communicate with other entities (agents, systems, humans) for its tasks.
In multi-agent systems, a large number of actors are able to interact. Local agents focus on the interests of local sub-systems and influence the whole system via negotiations with other software agents. While the complexity of individual agents remains low, the intelligence level of the global system is high. In this way, multi-agent systems implement distributed decision making systems in an open, flexible, and extensible way. Communication between actors can be minimized to a generic and uniform information exchange.

2.1.1 Electronic Markets

The interactions of individual agents in multi-agent systems can be made more efficient by using *electronic markets*, which provide a framework for distributed decision making based on microeconomics. Microeconomics is a branch of economics that studies how economic agents (i.e., individuals, households, and firms) make decisions to allocate limited resources, typically in markets where goods or services are being bought and sold. One of the goals of microeconomics is to analyze market mechanisms that establish relative prices amongst goods and services and allocation of limited resources amongst many alternative uses [21]. Whereas, economists use microeconomic theory to model phenomena observed in the real world, computer scientists use the same theory to let distributed software systems behave in a desired way. Market-based computing is becoming a central paradigm in the design of distributed systems that need to act in complex environments. Market mechanisms provide a way to incentivize parties (in this case software agents), that are not under direct control of a central authority, to behave in a certain way [7, 27]. A microeconomic theory commonly used in MAS is that of general equilibrium. In general equilibrium markets, or exchange markets, all agents respond to the same price, that is determined by searching for the price that balances all demand and supply in the system. From a computational point of view, electronic equilibrium markets are distributed search algorithms aimed at finding the best trade-offs in a multidimensional search space defined by the preferences of all agents participating in the market. The market outcome is *Pareto* optimal, a social optimal outcome for which no other outcome exists that makes one agent better-off while making other agents worse-off.

2.1.2 Market-based Control

In *Market-based Control*, agents in a MAS are competing for resources on an equilibrium market whilst performing a local control task (e.g., classical feedback control of a physical process) that needs the resource as an input. For this type of MAS, it has been shown by formal proof that the market-based solution is identical to that of a centralized omniscient optimizer [2]. From the viewpoint of scalability and openness of the information architecture, this is an important feature. In the centralized optimization all relevant information (i.e., local state histories, local control characteristics, and objectives) needs to be known at the central level in order to optimize over all local and global control goals. While in the market-based optimization the same optimal solution is found by communicating uniform market information (i.e., market bids stating volume-price relations), running an electronic equilibrium market and communicating the resulting market price back to the local control agents. In this way, price is used as the control signal. It is important to note that, whether — in a specific application — the price has a monetary value or is virtual and solely used as a control signal depends on the particular implementation and on the business case behind the application.

In a typical application of market-based coordination, there are several entities producing and/or consuming a certain commodity or good2. Each of these entities is represented by a local control agent that communicates with a market agent (auctioneer). Each market round, the control agents create their market bids, dependent on their state history, and send these to the market agent. These bids are ordinary, or *Walrasian*, demand functions $d(p)$, stating the amount of the commodity the agent wishes to consume (or produce) at a price of p. The demand function is negative in the case of production. After collecting all bids, the market agent searches for the equilibrium price p^*, i.e., the price that clears the market :

$$\sum_{a=1}^{N} d_a(p^*) = 0 \tag{2.1}$$

where N is the number of participating agents and $d_a(p)$, the demand function of agent a. The price is broadcast to all agents. Individual agents can determine their allocated production or consumption from this price and their own bid.

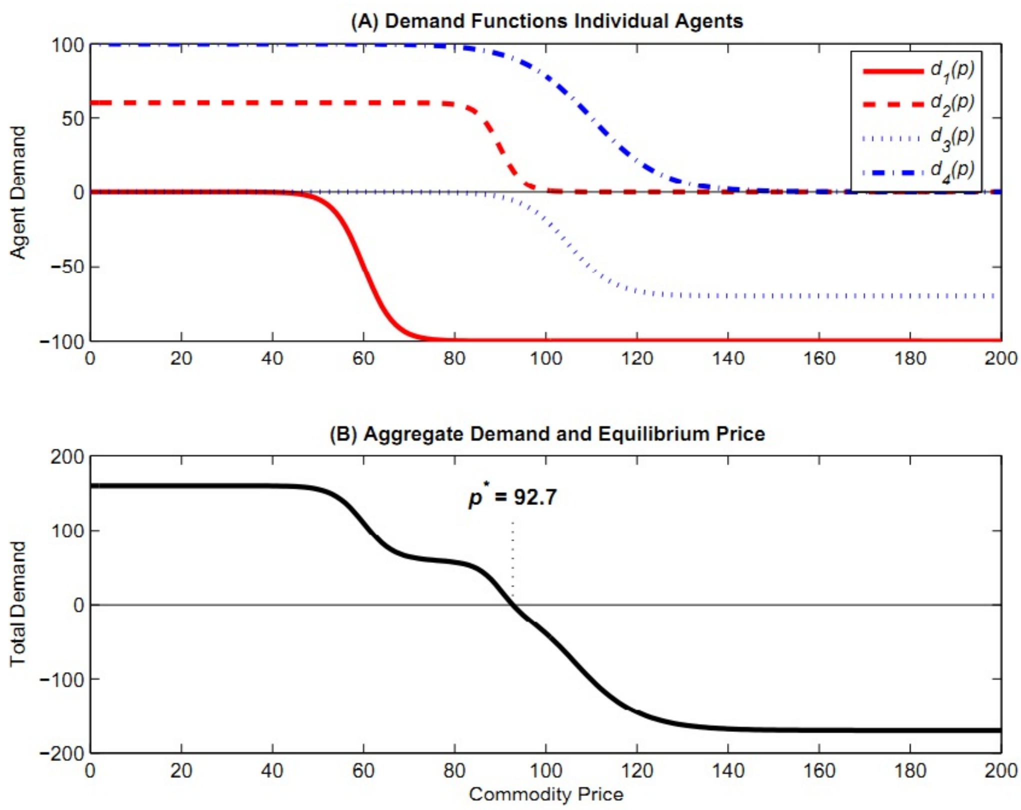

Figure 2.1: Example general equilibrium market outcome. (A) Demand functions of the four agents participating in the market. (B) Aggregate demand function and general equilibrium price p^*.

[2] Or a series of commodities. Here we treat the single-commodity case for simplicity

Figure 2.1 shows a typical small-scale example of price forming in a (single-commodity) general equilibrium market with four agents. The demand functions of the individual agents are depicted in graph (A). There are two consuming agents whose demand decreases gradually to zero above a certain market price. Further, there are two producers whose supply, above a certain price, increases gradually to an individual maximum. Note that supply is treated as negative demand. The solid line in (B) shows the aggregate demand function. The equilibrium price p^* is determined by searching for the root of this function, i.e., the point where total demand equals total supply.

2.1.3 Price-Based MBC: A Typical Example

In a typical price-based market-based control problem, there are several producing and/or consuming agents and an auctioneer agent. Each market round the producers and consumers create their market bids and send these to the market agent. These bids are ordinary, or *Walrasian*, demand functions $d(p)$, stating the agent's demand d at a price of p. The demand function is negative in the case of production. After collecting all bids, the market agent searches for the equilibrium price, i.e. the price at which the market clears. This price is broadcast to all agents, who can determine their allocated production or consumption from this price and their own bid. Finally, all producing agents feed their allocated production into the flow network while all consuming agents extract their consumption from it.

Figure 2.2 shows an example of price forming in a (single-commodity) general equilibrium market with four agents. The demand functions of the individual agents are depicted in graph (A). There are two consuming agents, whose demand decreases gradually to zero above a certain market price. Further, there are two producers whose supply, above a certain price, increases gradually to an individual maximum. Note that supply is treated as negative demand. In a control setting, the position of the inflexion point is typically determined by the current process state. The solid line in (B) shows the aggregate demand function. The equilibrium price p^* is determined by searching for the root of this function, i.e. the point where total demand equals total supply. The value of each agent's demand function at this prices is given in Table 2.1, Situation 1.

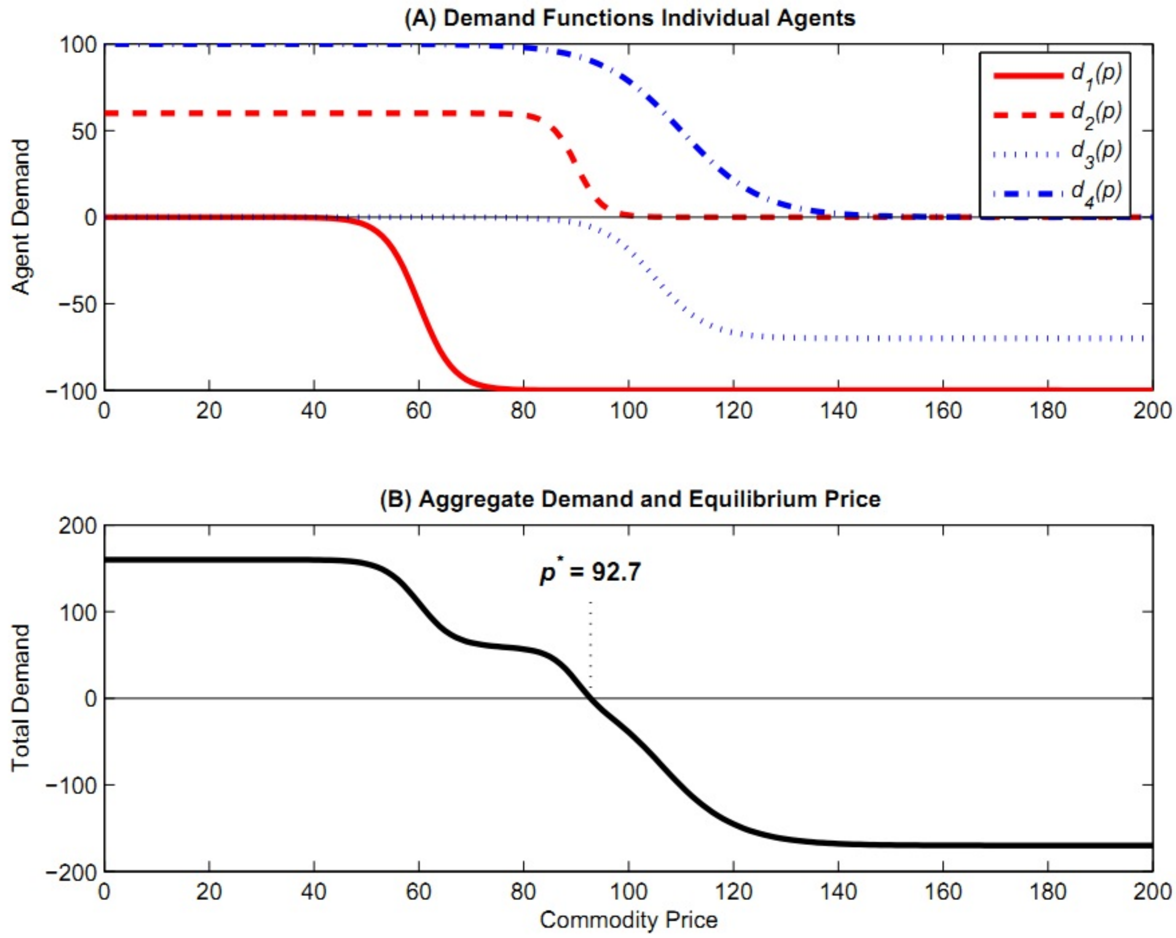

Figure 2.2: Example general equilibrium market outcome. (A) Demand functions of the four agents participating in the market. (B) The aggregate demand function. At price *p**, the market is in equilibrium: the sum of all supply and demand equals to zero.

Suppose the commodity traded in this example is electrical power. Suppose further, the first agent is associated with a unit for combined heat and power generation (CHP), e.g. used to heat a swimming pool. While serving the local heat demand, the unit produces electricity at the same time. Its local control goal is to keep a large water-filled heat buffer between two temperature limits. This buffer serves heat demand coming from subsystems such as space heating and heating of pool water. In the situation depicted by Figure 2.2, the CHP unit runs at full capacity. Its produced electricity is consumed by the two consuming agents and its produced heat is heating up the buffer.

Table 2.1: Agent demand levels for the two situations described in the text. Situation 1 corresponds to Figure 2.2, situation 2 to Figure 2.3.

	p^*	$d_1(p^*)$	$d_2(p^*)$	$d_3(p^*)$	$d_4(p^*)$	$\sum d_\alpha (p^*)$
Situation 1	92.7	-99.99	15.15	−5.56	90.41	0.00
Situation 2	109.8	0.00	0.02	−50.63	50.61	0.00

Suppose that sometime later, the heat buffer temperature is approaching the upper temperature limit. Then, the agent's need to produce heat — and, thus, its willingness to deliver electricity to the other agents — will be much lower. Now, the agent wants to produce electricity only if it gets a really good price for it and updates its bid accordingly. Figure 2.3 and Table 2.1, Situation 2, show the new situation. Due to the change in demand function of the first agent, the equilibrium price rises to 109.8. This causes the consuming agents to lower their intake, for agent 2 virtually to zero. The resulting demand is met entirely by the production of agent 4.

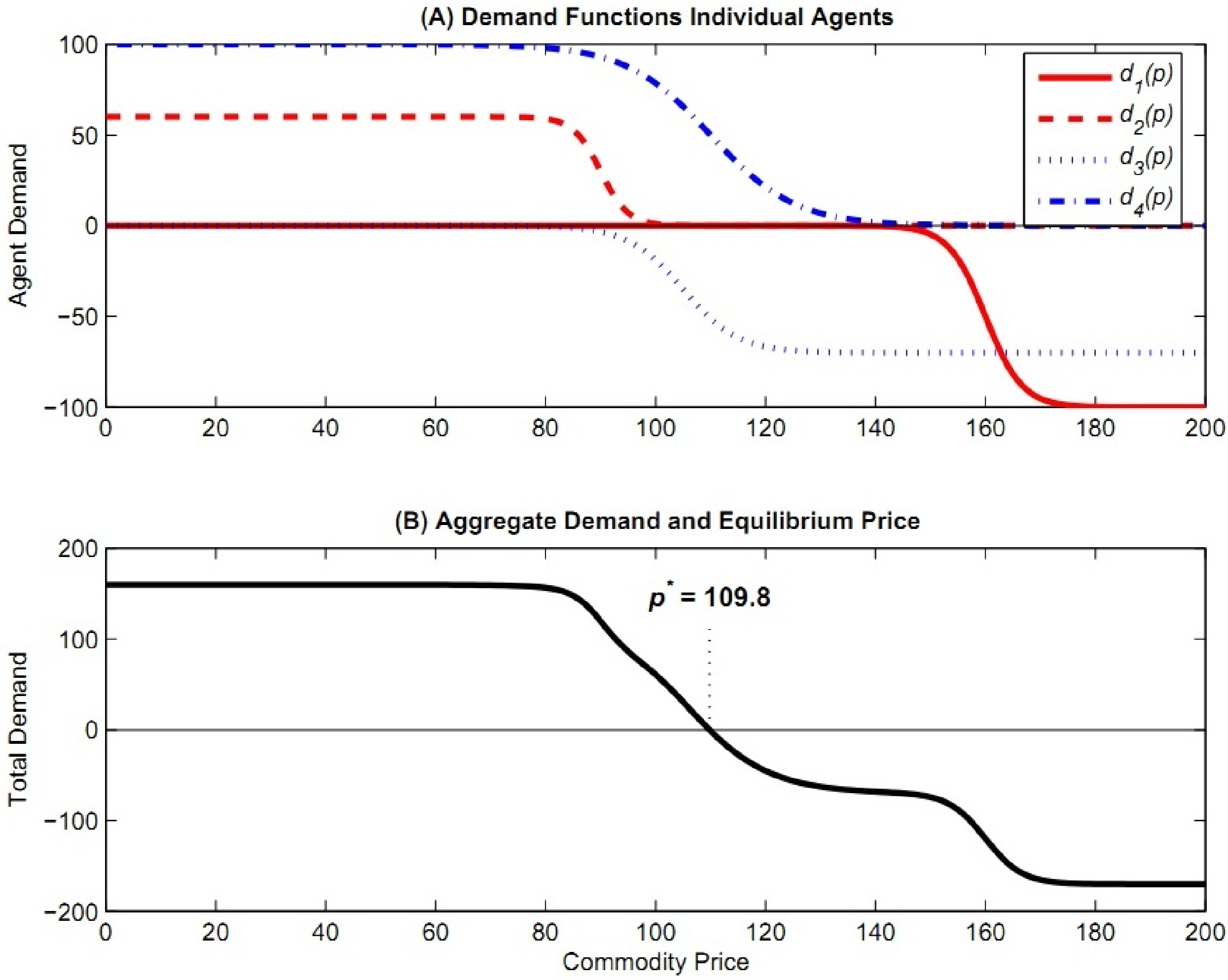

Figure 2.3: New market equilibrium after a change in the demand function of agent 1.

2.2 A Decentralized Control Systems Design

In earlier work, we designed a novel control concept for automatic matching of demand and supply in electricity networks with a high share of distributed generation. In this concept, DG, demand response, and electricity storage are integrated using the advanced ICT technology of market-based distributed control. This concept has been coined *PowerMatcher*.

Since its incarnation in 2004, the PowerMatcher has been implemented in three major software versions. In a spiral approach, each software version was implemented from scratch with the first two versions being tested in simulations and field experiments [18, 17, 26, 31]. The third version is planned to be deployed in a number of field experiments [25] and real-life demonstrations with a positive business case.

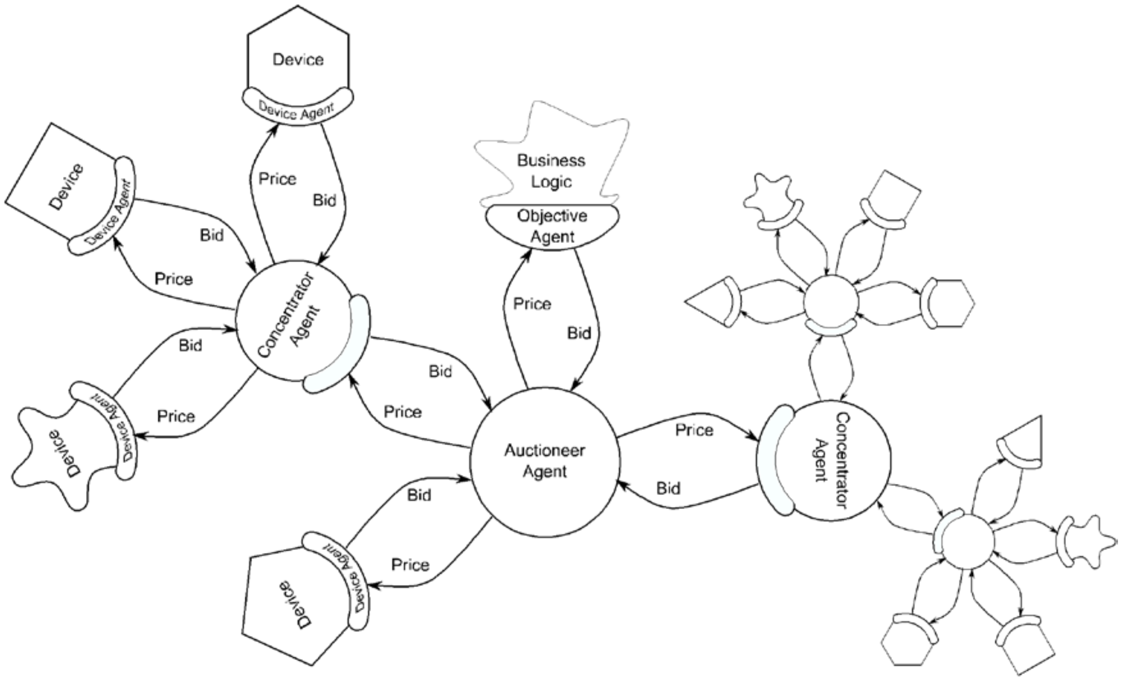

Figure 2.4: Example PowerMatcher agent cluster. See the text for a detailed description.

2.2.1 Logical Structure and Basic Agent Roles

Within a PowerMatcher cluster, the agents are organized into a logical tree. The leaves of this tree are a number of *local device agents* and, optionally, a unique *objective agent*. The root of the tree is formed by the *auctioneer agent*; a unique agent that handles the price forming by searching for the equilibrium price. In order to obtain scalability, *concentrator agents* can be added to the structure as tree nodes. More detailed descriptions of the agent roles are as follows:

- **Local device agent:** Representative of a DER device. A control agent which tries to operate the process associated with the device in an economical optimal way. This agent coordinates its actions with all other agents in the cluster by buying or selling the electricity consumed or produced by the device on an electronic market. In order to do so, the agent communicates its latest bid (i.e., a demand function) to the auctioneer and receives price updates from the auctioneer. It uses this received price, together with its latest bid, to determine the amount of power the agent is obliged to produce or consume.

- **Auctioneer agent:** Performer of the price-forming process. The auctioneer concentrates the bids of all agents directly connected to it into one single bid, searches for the equilibrium price and communicates a price update back whenever there is a significant price change.

- **Concentrator agent:** Representative of a sub-cluster of local device agents. It concentrates the market bids of the agents it represents into one bid and communicates this to the auctioneer. In the opposite direction, it passes price updates to the agents in its sub-cluster. This agent uses ‘role playing’. On the auctioneer’s side it mimics a device agent: sending bid updates to the auctioneer whenever necessary and receiving price updates from the auctioneer. Towards the sub-cluster agents directly connected to it, it mimics the auctioneer: receiving bid updates and providing price updates.

- **Objective agent:** Agent that gives a cluster its purpose. In absence of an objective agent, the goal of the cluster is to balance itself, i.e., it strives for an equal supply and demand within the cluster itself. Depending on the specific application, the goal of the cluster may be different. If the cluster has to operate as a *virtual power plant*, for example, it needs to follow a certain externally provided setpoint schedule. Such an externally imposed objective can be realized by implementing an objective agent. The objective agent interfaces the agent cluster to the *business logic* behind the specific application.

The logical agent structure follows the CoTree algorithm [34]. By aggregating the demand functions of the individual agents in a binary tree, the computational complexity of the market algorithm becomes $O(\lg a)$, where a is the number of device agents. In other words, when the number of device agents doubles it takes only one extra concentrator processing step to find the equilibrium price. Furthermore, this structure opens the possibility for running the optimization algorithm distributed over a series of computers in a network complimentary to power systems architectures. We discuss the issue of scalability further in section .

2.2.2 Basic Device Agent Functionality

For a DER unit to be able to participate in a PowerMatcher cluster, its associated agent must communicate its momentary *bid curve* or *demand function* to the Auctioneer. As described before, this function defines the DER's electricity demand $d(p)$ for a given price p. An offer to produce a certain amount of electricity against a certain price is expressed by negative $d(p)$ values. As a convention, throughout this text we refer to these functions as a bid, even when (part of) the function expresses a production offer.

Lets's focus on an agent for an electricity-consuming device, say a freezer. A simple block model of the thermal process of a freezer cell and it's external influences is depicted in Figure 2.5. Input to the process model is the boolean control variable $\alpha_{on/off}$, switching the freezing element on or off. Further, the temperature in the freezing cell is influenced by two environment variables: the ambient temperature (T_{amb}) and a usage pattern (ρ_{usage}). The latter represents usage events like door opening & closing and goods being placed in or removed from the cell.

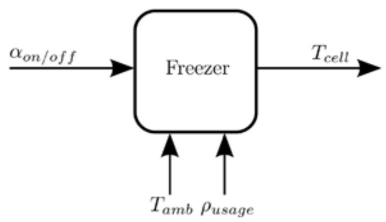

Figure 2.5: Freezer block model

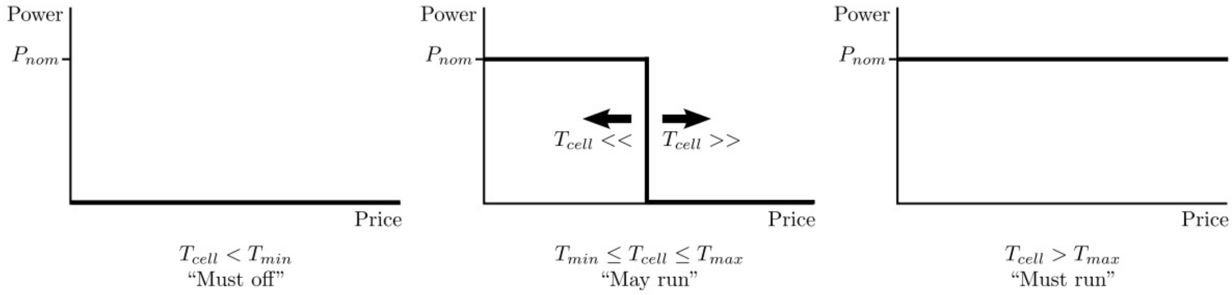

Figure 2.6: Three basic demand functions of a freezer.

The control goal is to keep the inner cell temperature within the temperature band given by: T_{max} and T_{min}, the maximum inner cell temperature and the minimum inner cell temperature, respectively. In a conventional freezer, this is achieved by a standard on/off-controller with hysteresis. When participating in a PowerMatcher cluster, this conventional controller is replaced by a device agent. The goal of the agent is, again, to keep the cell temperature between the given limits, with an additional goal to consume in low-priced periods as much as possible.
Figure 2.6 gives the three basic bid shapes for the freezer. When the cell temperature is below its minimum (*left*), the freezing element must be switched off. Accordingly, the device agent sends a *Must Off* bid. Similarly, when the cell temperature is above its maximum (*right*), the agent sends a *Must On* bid. The agent is forced to accept any price in order to get the cell temperature back within its limits. When the cell temperature is within limits (*middle*), the agent has the flexibility to switch on or off the element dependent on the electronic market price. Since the freezer element can either be switched on or off the agent's bid is a step function: bidding either for the freezer's nominal power or for a power of zero. The position of the step flank reflects the agent's willingness to pay. When the cell temperature is still in the lower part of the temperature band, the agent is only willing to consume when the price is really low. However, when the temperature rises, the agent's willingness to pay increases with it. So, available flexibility is directly dependent on the device state (here the cell temperature), and the position of the step flank in the agent's bid directly reflects that. In order to optimize its strategy, the agent needs to have market-knowledge, as the notion of what defines a "high price" or a "low price" is crucial in the agent's bidding strategy. We will come back to this aspect in the chapter about agent strategies.

2.2.3 Auctioneer and Concentrator Functionality

The core functionality of the auctioneer and the concentrators is to run the electronic market allocating the electrical power resource to the local device agents. The electronic market solves this allocation problem by finding the general equilibrium price p^* such that:

$$\sum_{a=1}^{N_a} d_a(p^*) = 0 \tag{2.2}$$

where N_a is the number of local device agents and $d_a(p)$ the demand function of agent a, stating the agent's demand or supply at a given price p.
The task of summoning all device agent's demand functions is divided over all concentrator agents and the auctioneer agent, here jointly referred to as *market agents*. Each market agent k summons the demand functions received from their attached agents. These functions originate from two different sources: (1) the device agents directly attached to k, and (2) the concentrator agents directly attached to k. The concentrated bid of k is calculated as:

$$a_k(p) = \sum_{j:x_j \in X_k} d_j(p) + \sum_{i:i \in Y_k} a_i(p) \tag{2.3}$$

where X_k is the set of local device agents directly connected to k and Y_k is the set of concentrator agents directly connected to k.

If k is a concentrator agent, it passes $a_k(p)$ on to the higher-level market agent it is attached to. If k is the auctioneer, it uses $a_k(p)$ to find the equilibrium price p^* such that the market is in equilibrium:

$$a_k(p^*) = 0 \tag{2.4}$$

Note that, in the latter case, a_k is the concentrated demand functions over all device agents:

$$a_k(p) = \sum_{a=1}^{N_a} d_a(p) \tag{2.5}$$

and that substitution of (2.5) in (2.4) yields the general market equation (2.2).

2.2.4 Classification of DER Controllability

From the viewpoint of supply and demand matching, DER devices can be classified in six classes according to their controllability characteristics. Below we describe each class and the basic agent strategy associated with it:

- **Stochastic operation devices:** devices such as solar and wind energy systems of which the power exchanged with the grid behaves stochastically. In general, the output power of these devices can't be controlled, the device agent must accept any market price.
- **Shiftable operation devices:** batch-type devices whose operation is shiftable within certain limits, for example (domestic or industrial) washing and drying processes. Processes that need to run for a certain amount of time regardless of the exact moment, such as assimilation lights in greenhouses, ventilation systems in utility buildings and circulation pumps in swimming pools. The total demand or supply is fixed over time. This class consists virtually only of electricity consuming devices. The agent strategy is to shift electricity consumption to time periods of low(er) prices.
- **External resource buffering devices:** devices that produce a resource, other than electricity, that are subject to some kind of buffering. Examples of these devices are heating or cooling processes, whose operation objective is to keep a certain temperature within two limits. By changing the standard on/off-type control into price-driven control allows for shifting operation to economically attractive moments, while operating limits can still be obeyed (see Figure 2.7). Devices in this category can both be electricity consumers (electrical heating, heat pump devices) and producers (combined generation of heat and power).
- **Electricity storage devices:** conventional batteries or technologies such as flywheels and super-capacitors coupled to the grid via a bi-directional connection. Grid-coupled electricity storage is widely regarded as a future enabling technology allowing the penetration of distributed generation technologies to increase at reasonable economic and environmental cost. Grid-coupled storage devices can only be economically viable if their operation is reactive to a time-variable electricity tariff, as is present in the PowerMatcher concept. The agent bidding strategy is buying energy at low prices and selling it later at high prices.
- **Freely-controllable devices:** devices that are controllable within certain limits (e.g., a diesel generator). The agent bidding strategy is closely related to the marginal costs of the electricity production.

- **User-action devices:** devices whose operation is a direct result of a user action. Domestic examples are: audio, video, lighting, and computers. These devices are comparable to the stochastic operation devices: their operation is to a great extent unpredictable and has no inherent flexibility. Thus, the agent must accept any market price to let them operate.

In all described device categories, agent bidding strategies are aimed at carrying out the specific process of the device in an economically optimal way, but within the constraints given by the specific process.

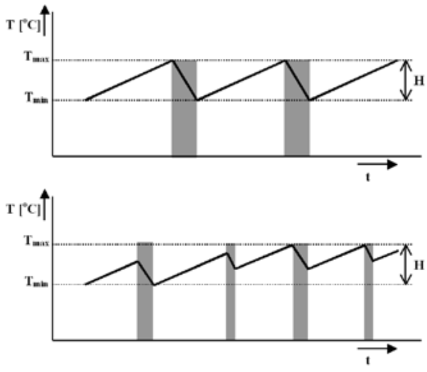

Figure 2.7: Operation shifting in a cooling process whilst obeying process state limits.

2.3 Design Choices

2.3.1 Communication Timing

The agents communicate in an event-based manner. Device agents update their bids whenever there is a change in the system state significant enough to justify a bid update. Typically, device agents update their bid once every few minutes or longer. Concentrators, in turn will not update their bid unless subsequent updated bids from lower agents result in a significant change in their concentrated bid. Likewise, the auctioneer will only communicate a new price after a considerable price change. In this way, coordination on a timescale of minutes is realized with low volumes of communicated data. For the two main application cases of the PowerMatcher, commercial portfolio balancing and congestion management , this type of *near real-time coordination* suffices, as these processes take place on a similar timescale.

2.3.2 Design for Scalability

In the design of the PowerMatcher a number of choices have been made to meet the important requirement of scalability. The three main scalability choices are: the use of a *pool market*, *one-shot* communications and *distributed aggregation* of demand functions.

Pool Market vs Peer-To-Peer Trading

Imagine a market where people come together to buy and sell apples. People that go there to buy apples want to buy their apples for a good and fair price. So, before they buy their apples, they ask around among sellers what their price for an apple is. Further, they exchange with other buyers information on the bargains. The sellers, on the other hand, try to sell as much apples as possible for a price highest as possible. However, sellers that ask too high a price won't sell too much apples and do not earn much money. Sellers that ask to low a price sell a lot of apples, but could earn more money when they would ask a bit more. To find the right selling price, sellers look around to find out what price the competitors are asking and talk to or negotiate with

buyers to get an idea about their willingness to pay. The price on such a market evolves to an equilibrium price (p^*), the 'going' price for an apple. As this price is a general price for the market as a whole, the market is said to be in *general equilibrium*.
This general equilibrium can be reached in the peer-to-peer manner as described above. Then, the exact equilibrium price would be reached when all buyers negotiate with all sellers and, thus, everyone has complete information. On the other hand, an auctioneer could act as a market operator. Then, all buyers and sellers communicate with the auctioneer only. The auctioning process starts with the auctioneer calling off a price. Then, all buyers and sellers state to the auctioneer the number of apples they are willing to sell or buy for that price. The auctioneer sums up all amounts to see if the market clears. The auctioneer calls off a higher price in case of excess demand, and a lower price if there is excess supply. The auctioneer iterates through this process until the market-clearing price p^* is found. Note that the market outcome is equal to the case in which all participants hold complete information, however, without the necessity for each participant to communicate with each of the others.

Trusted Auctioneer: One-shot Communications

Note that, in the case described above, each buyer or seller a needs to have his own demand function $d_a(p)$ in mind. When the auctioneer calls off price p_x, each buyer and seller states his preferred amount of apples for that price, given by $d_a(p_x)$. Note further, the auctioneer has to be trusted by all actors participating in the pool market in order to let him play the role as a middleman. When the auctioneer is trusted indeed, the number of communication steps between auctioneer and all participants can be reduced drastically if the full demand functions are communicated at once. Then, the iterative process of finding the clearing price by the auctioneer does not include any communication with participants any more. The whole process reduces to a *one-shot* communication of $d_a(p)$ of all a to the auctioneer, followed by a communication-free clearing price search by the auctioneer and again a one-shot communication of the resulting price p^* to all participants.

Distributed Concentration of Demand Functions

Introducing one-shot communications drastically limits the number of communication steps in the process. However, now, the auctioneer is the hub in the electronic market wheel. All demand functions need to be communicated to one single point in order to run the market. When the number of agents participating in the market grows further, this system again runs into a communication complexity problem when the auctioneer can't handle all communications fast enough. The solution to this problem lies in the electronic market algorithm. The price search involves the summation of all $d_a(p)$ into a concentrated demand function $\sum$ da (p) and finding the equilibrium price p^* for which this concentrated function equals to zero: $\sum d_a(p^*) = 0$. The calculation of the concentrated bid and the subsequent communications can be distributed over a number of concentrator agents. Then, a number of concentrator agents collect the demand functions of a mutually exclusive subset of market participants and calculates the concentrated bid for this subset. The result is communicated further toward the auctioneer. At the top of the structure, the auctioneer does the last concentration step and searches for the equilibrium price.
Imagine a market with 1 million market participants and a market structure having an auctioneer and two layers of concentrators of 100 and 10,000 pieces respectively. The auctioneer and each of the concentrator agents communicate with 100 agents in the layer directly below it, which is a low complexity communications task. Further, concentration of bids happens in parallel within each concentrator layer. When the number of market participants doubles, the whole structure below the auctioneer is duplicated and one extra concentrator is added. This hardly adds to the overall computation and communication complexity.

3. Individual Agent Strategies

A key activity of a PowerMatcher cluster of agents is the delivery of near real-time balancing services, e.g. delivering reserve regulating power to the TSO, delivering active network management services to the DSO or minimizing the imbalance costs of a commercial party. In order to operate such a near real-time coordination activity optimally, the agent society maintains a dynamic merit-order list of the (typically large number of) DER units participating. To make optimal decisions based on this list, the merit order needs to be based on the true marginal cost (or marginal benefit in the case of demand response) of the individual DER units. The marginal electricity costs of most types of DER are highly dependent on local context and, hence, change over time. For example, the marginal electricity production cost for a CHP is highly dependent on the amount of heat demanded from the unit at a particular time. Thus, when the heat demand is high, the marginal cost for the electricity production is low and vice versa. The dynamic marginal cost levels of the units in the cluster result in the dynamic nature of the merit order list. As we will show later, there exists a class of DER units for which, under circumstances, the marginal cost level can't be determined unambiguously.

From a micro-economic viewpoint, the DER units are assumed to participate in a competitive market. This assumption holds when the number of DER units in the agent society is relatively high and their traded volumes are of the same order of magnitude. A competitive market leaves no room for speculation or gaming, and the best (*i.e.* the *dominant*) strategy for each participant is to optimize its own utility by truly bidding its marginal cost [21]. These locally-optimal strategies lead to a merit order list that results in an optimal allocation on the global level as well, as those DER which are best fit to respond to a certain event are the first to be selected to do so.

In this chapter, we investigate the mechanisms that determine the momentary marginal costs of distributed generators and the momentary marginal benefits of demand response resources. The existence of a bid strategy spectrum is shown. At the end we discuss an example of a small island grid.

3.1 Agent strategies based on short-term economics

As described in the chapter introduction, the optimal strategy of an agent active on a competitive market is to bid according to its momentary marginal cost. For a PowerMatcher device agent, the bidding strategy is a mapping from its context history to a market bid. This context includes:

- The process controlled by the agent, including the current state of the process and economical parameters such as marginal operating cost.
- The market environment in which this agent is situated, including the market mechanism and market prices.

In the extremes, there are two agent types that are forced to base their bid on either of the two context elements described above:

1. Those agents operating a DER unit that has clear and unambiguous levels of marginal costs. In a competitive market, the dominant strategy of these agents is to bid entirely according to their marginal operating costs.

2. Those agents operating a DER unit that does not have unambiguous marginal costs at all. In these cases, the bidding strategy can only be based on market parameters, i.e. the market price (history).

As said, these cases are the extremes of a spectrum and hence, there is a group of agents whose bidding strategy is somewhere in between these extreme cases. In the next subsections, we will give examples of these extreme and median cases.

3.1.1 A Strategy Fully Based on Marginal Cost

An example of a bidding strategy entirely based on the marginal cost level is that of a fuelled electricity generator set, for instance a gas generator set. The marginal cost for a given period of operation depends on the fuel price, the efficiency of the generator and the running history dependent maintenance costs. Furthermore, each startup of such a generator causes additional costs for maintenance and fuel. The dominant strategy in this case is bidding a price equal to the marginal operation cost.
The bidding strategy is a function of the following parameters:

p_f	[ct/m^3]	Fuel price
r_g	[Wh/m^3]	Generator fuel rate
P_g	[W]	Generator electrical power
m_r	[ct/h]	Maintenance cost rate
c_s	[ct]	Additional start-up maintenance costs
f_s	[m^3]	Additional start-up fuel use

The marginal cost for operating the generator for a time period of Δt is:

$$c_{m,r}(\Delta t) = \left(\frac{P_g p_f}{r_g} + m_r\right)\Delta t \quad (3.1)$$

$$c_{m,s}(\Delta t) = c_{m,r}(\Delta t) + c_s + f_s p_f \quad (3.2)$$

where $c_{m,r}$ is the marginal cost when the generator is already running at the start of the Δt time period, and $c_{m,s}$ when it has to be started up. Therefore, the optimal bidding function is given by:

$$d(p) = \begin{cases} 0 & \texttt{if } p < c_m \\ -P_g & \texttt{Otherwise} \end{cases} \quad (3.3)$$

where c_m equals either $c_{m,r}$ or $c_{m,s}$ depending on the running state of the generator. Note that, by definition, $d(p)$ is negative in case of supply, hence the minus sign before the P_g term. It is clear that this bidding strategy depends entirely on the cost parameters of the generator. The market price history does not play a role in this strategy.

3.1.2 A Strategy Fully Based on Price History

At the other extreme is the bidding strategy of an electricity storage facility. Systems such as batteries, flywheels and pumped storage, charging from the electricity grid at one time and discharging to it at another. The aim of the agent is to buy electricity in periods of low prices, store it and resell in periods of high prices. Hence, the notion of what defines a “high price” or a “low price” is crucial in the agent's bidding strategy. Maximizing the agent's utility comes down to determining the charge/discharge price that yields the best profit. This *optimal price set* is entirely dependent on the dynamic price characteristics of the market environment plus the time needed for a full charge or discharge.

Charging and discharging a storage device is subject to *round-trip energy losses*. Note that, for the operation of a storage system to be profitable in the long run, the margin between the buy price and the resell price must exceed the costs for these losses. However, these costs do not influence the optimal price levels themselves.

Therefore, the agent requires some sort of function E that yields estimates of the optimal charge and discharge prices given the current price history and the charging/discharging time:

$$< \overline{p}_c, \overline{p}_d > \;=\; \mathcal{E}(H_p, T_s) \quad (3.4)$$

$$T_s = C_s / P_s \quad (3.5)$$

where:

P_s	[W]	Storage charging/discharging power
C_s	[Wh]	Storage capacity
T_s	[h]	Storage charging/discharging time
H_p	[ct]	Price history vector

Based on these estimated price levels the bidding function can be defined by:

$$d(p) = \begin{cases} P_s & \texttt{if } p < \overline{p}_c \\ -P_s & \texttt{if } p > \overline{p}_d \\ 0 & \texttt{Otherwise} \end{cases} \quad (3.6)$$

The long-run profit is highly dependent on the quality of the estimator $\overline{E}$, which must operate in dynamic market environments whose characteristics, in most cases, will be unknown at design time.

3.1.3 A Median Strategy

This case is based on configurations found in installations supplying heat to residential areas: a CHP/Gas heater combination. A typical configuration combines a CHP, a more traditional gas heater and a heat storage buffer. An installation of this type was part of one of the earlier field trials with the PowerMatcher.

The marginal cost levels depend on the following parameters:

η^t_{chp}	[]	Thermal efficiency of the CHP
η^e_{chp}	[]	Electrical efficiency of the CHP
η^t_{htr}	[]	Thermal efficiency of the heater
P_g	[ct/m^3]	Gas price
H_c	[kJ/m^3]	Gas combustion heat
T_{max}	[^{o}C]	Upper limit inner temperature heat buffer
T_{min}	[^{o}C]	Lower limit inner temperature heat buffer

Typically, the thermal efficiency of the heater will be higher than that of the CHP: $\eta^t_{htr} > \eta^t_{chp}$.

The heat demanded by the residential area is subtracted directly from the heat buffer. The local control goal of the CHP/heater combination is to keep the inner temperature of the buffer, T, between thermal limits T_{max} and T_{min}. Hence, the buffer level is defined as:

$$L_B = \frac{T - T_{min}}{T_{max} - T_{min}} \tag{3.7}$$

To prevent the buffer from over or under heating, three levels are defined at which special control actions are to be taken:

- L_H : High buffer level: just below the fill level of 100%. Above this level both the CHP and the heater must be switched off to prevent overheating. CHP operation is only possible in combination with heat dump, if that is technically possible (and ethically acceptable).
- L_L: Low buffer level: the level under which either the heater or the CHP must be switched on to prevent under heating.
- L_{LE} : Low emergency level: just above 0%. Below this level both heater and CHP must be switched on.

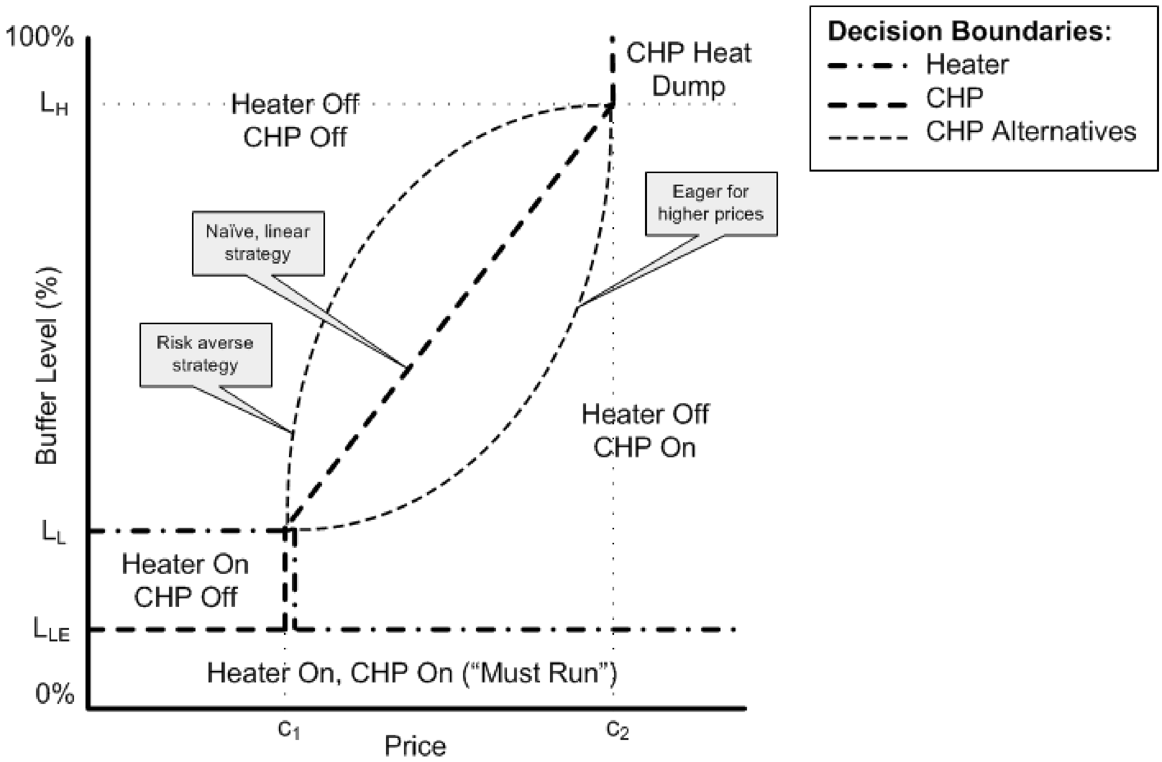

Figure 3.1: Bid strategy of a Heater/CHP combination as found in heat network systems delivering heat to residential areas. The strategy is well-defined below c1 , the marginal cost for CHP-produced electricity when heat demand is high, and above c_2, the CHP's marginal electricity cost when there is no heat demand at all.

These levels define four different operational modes (see figure 3.1):

1. **Below L_{LE}** , the high heat demand is the dominant factor in the operation of the installation. This is a must-run situation for both CHP and heater, regardless of the electricity price.

2. **Between LLE and LL**, there is a heat demand that could be met by either the heater or the CHP.

Hence, there is a choice of producing this heat using the heater or the CHP. In the latter case, the operating costs will be higher (as $\eta^{t}_{chp} < \eta htr$) with additional electricity production in return. While the heat demand is covered by the CHP, the marginal cost of the additional electricity production is equal to:

$$c_1 = c^t_{chp} - c^t_{htr} \tag{3.8}$$

where c^t_{chp} is the marginal cost for heat produced by the CHP regardless the value of the co-produced electricity and c^t_{chp} is the marginal cost for the heater-produced heat.
With:

$$c^t_{chp} = \frac{p_g}{H_c}\eta^t_{chp} \tag{3.9}$$

$$c^t_{htr} = \frac{p_g}{H_c}\eta_{htr} \tag{3.10}$$

equation (3.8) can be expanded to:

$$c_1 = \frac{p_g}{H_c}(\eta^t_{chp} - \eta_{htr}) \tag{3.11}$$

Accordingly, the CHP is operated when the market price for electricity is higher than c_1, otherwise the heater is operated.

3. **Above buffer level LH**, there is no heat demand. Hence, there is a choice to run the CHP and dump the produced heat. Even if the installation is not technically capable to discard CHP-produced heat, the marginal cost level of this option is of interest as it provides one of the strategy boundaries of the forth operation mode, described below.
   During CHP operation just for electricity production, the marginal cost for the electricity equals:

$$c_2 = \frac{p_g}{H_c}\eta^e_{chp} \tag{3.12}$$

   If the market price is above c_2, it is profitable to run the CHP, even when the produced heat is discarded.

4. **In the region between L_L and L_H,** there is a high level of freedom to let the CHP run dependent on the electricity price. At both boundaries of this region, the bidding strategy is well defined: at level L_L it is profitable to produce whenever $p > c_1$, while at level L_H it is profitable to produce whenever $p > c_2$. The 'naive' or 'ignorant' strategy would be to connect these two points linearly. However, dependent on both the dynamic price characteristics of the market *and* the risk profile used, different trajectories are possible. In figure 3.1, two alternative strategies are shown. The risk-averse strategy tries to avoid must-run situations for both CHP and heater by taking the chance to fill the buffer whenever it is profitable to run the CHP. The other alternative strategy waits for higher prices to operate the CHP, with a higher risk of missing profit opportunities and ending in the must-run regions for heater and CHP.

3.2 Bid Strategy Spectrum

As becomes apparent, there exists a spectrum of DER bidding strategies. On one end of the spectrum, bidding strategies are based directly on true marginal cost or benefit. Along the spectrum, optimal bidding strategies become less dependent on marginal cost levels and more on the price dynamics in the (VPP) market context. As may be clear from the description of the CHP/Gas Heater combination,

price-dynamics based strategies are not unambiguously defined but are dependent on a desired risk level.
In figure 3.2, the relative positions of a number of DER units are shown. Below, we discuss briefly the spectrum position of units not described previously.

- Generators of renewable power, such as wind turbines and photo-voltaic solar systems, typically have low marginal costs associated with them, as these consist mainly of maintenance costs. Fuel costs, the main marginal cost component for most other generation types, are essentially absent here. Therefore, the dominant strategy of renewables is to generate at any going electricity price. This positions them at the marginal-cost based extreme of the spectrum.

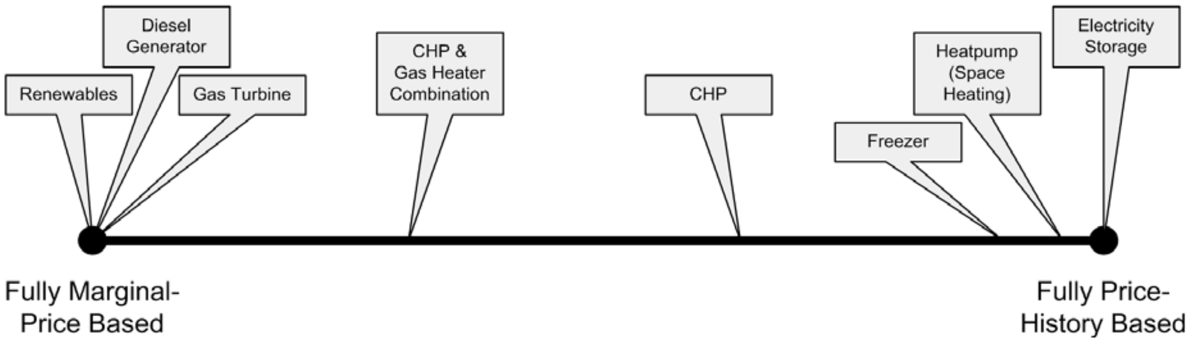

Figure 3.2: Bid Strategy Spectrum for Distributed Energy Resources based on momentary marginal cost levels.

- **CHP with heat buffer:** In high-price situations, the bidding strategy of a solitaire CHP is similar to that of the CHP/Heater combination. The marginal cost for CHP produced electricity in the (theoretical) heat-dump case (c_2 in figure 3.1) is applicable here as well. However, the low-price behavior is dependent on the value attached (by the user) to a reliable heat supply and the risk level one allows for occasionally not being able to cover the heat demand entirely. Minimizing this risk is highly dependent on the prevailing price-dynamic characteristics. Hence, the position of CHPs on the right-hand side of the spectrum.

- **Direct Electrical Space Heating or Cooling:** Modern building constructions show relatively high degrees of thermal inertness. This can give some degree of freedom in the operation of systems for space heating and cooling, but is dependent on the current temperature and the temperature desired by the user. As learnt in field experiences, it is possible to shift cooling or heating periods forward or backward in time without infringing user comfort [26, 31]. Here, the agent strategy goal is to provide the desired comfort level against minimal electricity costs, shifting cooling/heating actions towards low-priced periods as much as possible. Comparable to the strategy for storage units, the notion of what ’low prices’ actually are is crucial for a successful strategy. This locates this DER type directly in the price-history based end of the spectrum. However, as learnt from experiences with demand response programs aiming at influencing user behavior, most users are willing to offer some comfort in order to avoid periods of high tariffs. Due to this, we position Direct Electrical Space Heating or Cooling just left of the spectrum end.

- **Freezer:** The case of a freezer is similar to that of that of space heating/cooling described above, hence the position near the price-history based end of the spectrum. As a minor difference, for this instance, the cost of ’lost service’ is known as this equals the total value of the stored food items.

A bid strategy spectrum exists for DER units being part of a market-based control cluster delivering (near-) real-time balancing services. On one end of the spectrum, bidding strategies are based straightforwardly on true marginal cost or benefit. On the other spectrum end, optimal bidding strategies are dependent on the price dynamics in the electronic market context and the desired maximum risk level. These results are relevant both from business economic and technical perspectives:

- **Business economic relevance:** our results contribute to the understanding of the business economics of Virtual Power Plants and active distribution networks. A good understanding of marginal cost mechanisms of DER units participating in a VPP or active network gives insight in the profitability of these measures.
- **Technical relevance**: the technical challenge is to design agent societies that find an optimal division of work in a given cluster of distributed generators and demand response resources under all circumstances. As we have shown, the merit order in such a society is highly dependent on the local context of the DER units in the cluster. Insight in these dependencies is necessary to design optimal VPPs and active networks.

3.3 Example: a local island grid

Imagine a small, isolated island with a local electricity network with no connection to an outside electricity network. The village of this island has 10 houses. Half of the houses are heated by heatpumps, the other half by micro-CHPs. Apart from the heatpumps, the energy consumption within the houses is inflexible and following standard household load profiles. Further, on the island there is a wind-diesel combination delivering that part of the momentary electricity demand not supplied by the CHPs. This combined unit is operated to balance the island system. When the local demand is higher than the CHPs and wind turbine are producing, the diesel generator is regulated to maintain the momentary system balance. On the other hand, when local demand is lower than the CHP and wind generated power, the wind turbine is curtailed and regulated to balance the network.
In a small-scale simulation, the impact of the PowerMatcher was analyzed for the hypothetical island system described above. The simulation has been carried out for two distinct cases:

1. **Reference Case.** This is the business as usual scenario. The heating systems are controlled by a standard thermostat on/off controller. The system is balanced entirely by the wind-diesel system.

2. **Coordinated Case.** In this case the micro-CHPs and the heat pumps (HPs) are coordinated by the PowerMatcher. The multi-agent system tries to match CHP production and HP consumption with the inflexible demand and supply of the households and wind turbine respectively. Any net surplus or shortage is still balanced by the wind-diesel combination.

Table 3.1 gives the characteristics of the units used. The wind turbine output followed the measured production profile of a real-world turbine (Figure 3.3). The heating systems, i.e. the micro-CHPs and the heat pumps, were used for space heating alone. At this stage, hot tap water demand was left out of the scope of the simulation. The heat demand was generated using a basic thermal model of a house. The main external variable of this model is the outside temperature, which was set to follow a standard reference pattern. The household electricity consumption followed a standard residential load profile. Goal of the simulation is to give a proof of principle of the coordination mechanism, illustrating the cluster-level behavior.

Table 3.1: Electricity producing (P) and consuming (C) units in the island simulation. The flexible units can be coordinated by the PowerMatcher.

Type	P_{max}	Number	P/C	Flex?
Diesel generator	15 kW	1	P	yes
Wind Turbine	30 kW	1	P	no
Micro CHP	1 kW	5	P	yes
Heat pump	0.7 kW	5	C	yes
Household Load	1.1 kW	10	C	no

The simulation spans a period of two days. Figure 3.4 gives output power of the diesel generator in the two cases. Two important effects can be seen from the figure:

- The total production of the diesel generator is approx. 40% lower in the coordinated case.
- The peak load served by the diesel generator is approx. 45% lower in the coordinated case.

The first effect is an important result as the environmental footprint of the island's electricity system is improved. Apparently, the wind generated power is utilized better in the coordinated case. More wind power is consumed and the turbine has been curtailed less. The second effect is important from an investment point of view. If the peak load on the diesel system is lower, the unit's design capacity can be lower which leads to a lower investment.

Figures 3.5 and 3.6, show the temperatures in the rooms heated by the heat pumps. The local PowerMatcher device agents make use of the inherent energy buffer in the inner space of the houses to shift the heating operation. Note that at all times the comfort level is maintained. Figure 3.7 gives the price on the electronic market for the simulation period. Note that the device agents in figure 3.6 try to heat the homes in the low-priced periods. The resulting price is influenced by a number of factors: (1) the momentary wind power availability, (2) the momentary household electricity demand, (3) the available operational flexibility of the micro-CHPs and the HPs. Note further that the diesel generator is only operated in the high-priced periods. Then, the cluster can't provide the needed generation capacity, resulting in high prices and, in turn, utilization of the generator.
To summarize: the self-interested behavior of local agents causes electricity consumption to shift towards moments of low electricity prices and production towards moments of high prices. As a result, the emergence of supply and demand matching can be seen on the global system level. The aggregated, or concentrated, bid of all local control agents in the cluster —as held by the auctioneer agent— can be regarded as a dynamic merit-order list of all DER participating in the cluster. Based on this list, the units that are able to respond to a certain event most efficiently are selected to do so. In this way, the (near-)real-time coordination mechanism of the PowerMatcher lets the cluster as a whole operate optimally.

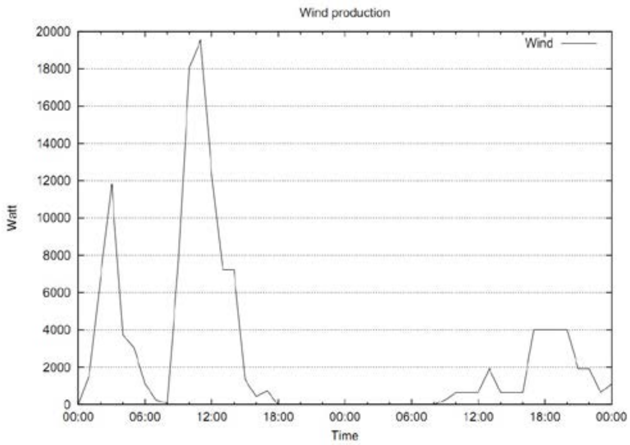

Figure 3.3: Power Output of the 30 kW wind turbine over the two-day simulation period.

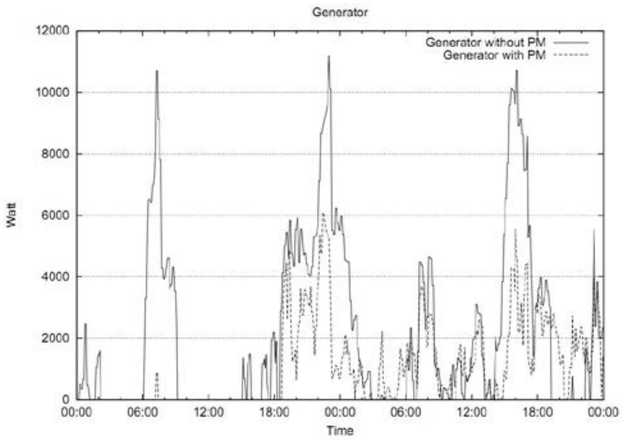

Figure 3.4: Diesel generator output power for the reference case (solid line) and the coordinated case (dashed line) over the two-day simulation period.

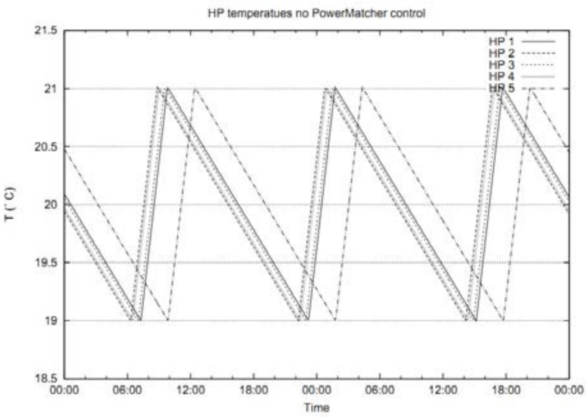

Figure 3.5: Room temperatures of the 5 heat pumps in the reference case. The basic On/Off controller behavior can clearly be seen.

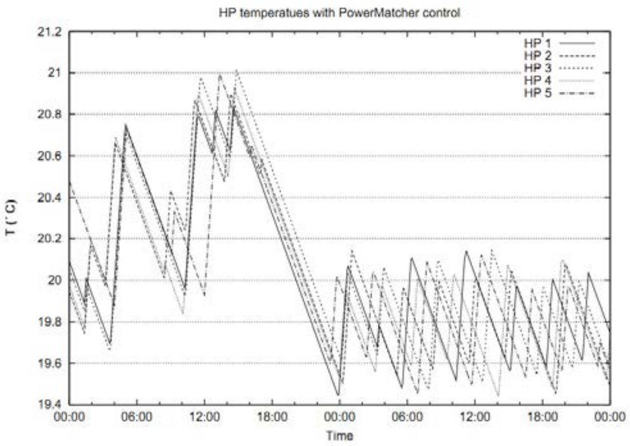

Figure 3.6: Room temperatures of the 5 heat pumps in the coordinated case. PowerMatcher control.

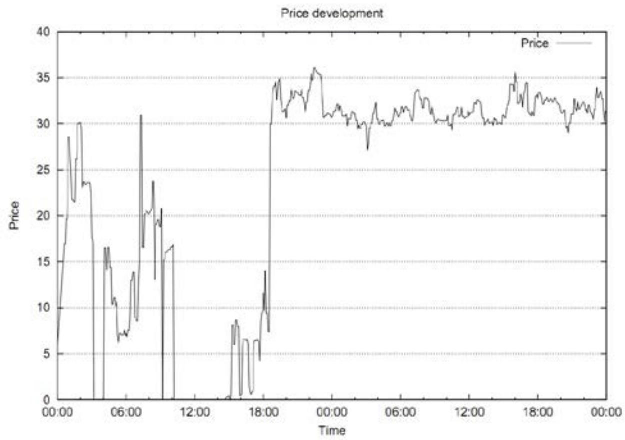

Figure 3.7: Price development of the PowerMatcher electronic market.

4. Transport Network Feasible Solutions

In multi agent resource allocation, major advances have been made towards algorithms with high scalability regarding both the number of participating agents and the number of commodities. However, the current market-based resource allocation algorithms for flow commodities do not take characteristics of the underlying transport network into account. For instance, capacity constraints in the transport network often have impact on the feasibility of a particular allocation outcome. By not considering these constraints in the market algorithm used, it is implicitly assumed that the network has virtually infinite capacity. Other network characteristics relevant to certain application cases are transport losses and changes in the amount of commodity stored in the transport network itself. The interaction between passive flow networks and allocation algorithms is not yet described from the viewpoint of market-based flow commodity allocation in multi agent systems.

On the other hand, power systems economics does provide a framework for market-based coordination in passive flow networks under network capacity constraints. This framework is known as *locational marginal pricing*.

In the work presented in this chapter we:

1. translate the framework of locational marginal pricing [28] from the field of power systems economics into computer science. In this reformulation, we:

   (a) omit modelling details oriented towards bulk transmission and wholesale trade of electricity;

   (b) bring the framework into multi-agent systems theory where agents communicate their preferences in the form of demand functions; and

   (c) generalize the framework to be applicable for *all* flow commodities, such as gasses, liquids and electricity.

2. show that, under the common condition of demand and supply elasticity, the constrained optimization problem posed by the framework has a unique solution and a search in the parameter space will converge to that solution; and

3. provide a distributed market algorithm that solves the constrained optimization problem.

The market algorithm can be regarded as a generalization of electronic equilibrium markets. Under network capacity constraints, it finds solutions that are feasible for the underlying passive flow network. In non-constrained networks, its solution is equal to the general equilibrium.

In generalizing the framework for all flow commodity types, price components have been included for transport losses and storage of the commodity inherently in the network itself. For material flows in the gas or liquid phase, transport losses are pressure-driven (e.g. filtration or permeation of pipes and connections). For electrical energy these are Ohmic losses: dissipation of electrical energy into heat in network components such as cables and transformers. Inherent storage applies mainly to gas flows:, the amount of material stored in the network is influenced by the average system pressure.

Section 4.1 gives an example of locational pricing in micro-economic control and gives a brief overview of related work. Further, the important conceptual difference between passive flow networks and actively switched networks is described here. Concepts of network flow modeling (Sec. 4.2.5) are important for the theoretical framework (Sec. 4.3) as well as the distributed market algorithm described in Section 4.5. The latter section also demonstrates the algorithm for a medium-sized network. Section 4.4 analyzes search space and convergence properties.

4.1 The Concept of Locational Pricing

4.1.1 “A Typical Example” Revisited

In section 2.1.3, we described an example of price forming in a single-commodity general equilibrium market with four agents. Here, we will revisit that example reproducing Figure 2.2 in the left part of Figure 4.1. In that figure, we showed the demand functions of the individual agents (graph A) and resulting equilibrium price (graph B).

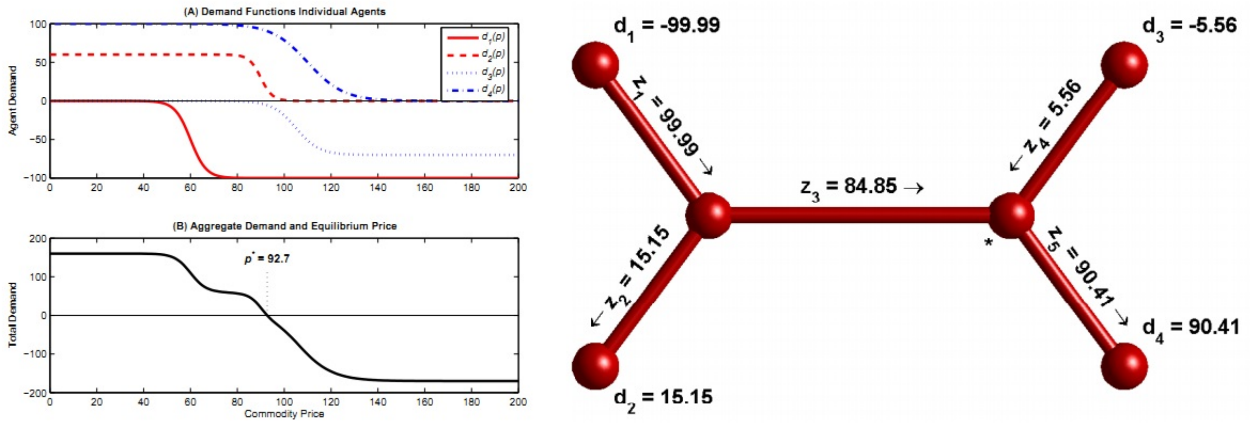

Figure 4.1: *Left:* Reproduction of Figure 2.2: Example general equilibrium market outcome. (A) Demand functions of four agents. (B) The aggregate demand function and resulting equilibrium price. *Right:* Resulting network flows through an H-shaped network when these agents are located in the endnodes. Note that supply is indicated by negative demand.

Now, suppose these agents are located the H-shaped flow network shown in the right part of the figure. Each of the four end nodes accommodates one of the agents. The individual agent demands (dk) at price p^*, that are subtracted from the network, are indicated at the corresponding nodes. The two intermediate nodes are just connection points and accommodate no additional demand or supply. Then, the resulting commodity flows through the network connections are as indicated in the figure. The subnetwork on the left-hand side is a net producer of the commodity which results in a strong flow to the net-consuming right part of the network.

Table 4.1: The network feasible line flows of the described example (with all flow directions equal to those in Figure 4.2).

z_1	z_2	z_3	z_4	z_5
85.00	60.00	25.00	38.07	63.07

Now, focus on line 3, connecting the left and right parts of the network. The price reaction of the left and right subnetwork are respectively given by:

$$d_{left}(p) = d_1(p) + d_2(p) \tag{4.1}$$

$$d_{right}(p) = d_3(p) + d_4(p) \tag{4.2}$$

In Figure 4.2 (Graph B), these two aggregate demand functions are added to the original figure . Note that, per definition, $d_{right}(p^*) = -d_{left}(p^*)$, and $|z_3| = |d_{left}(p^*)|$. This is indicated in the figure by the vertical black line at $p*$. In words, at the equilibrium price, the excess supply in the left subnetwork equals the excess demand in the right subnetwork, and is equal to the flow through the interconnection.

Now, suppose line 3 has a maximum capacity: $z_{3,MAX} = 25$, in which case the general equilibrium solution overloads this line more than threefold. Any market-based method to solve this constraint violation has to follow the general modus operandi of market-based systems (as described in section 2.1), i.e. to incentivize actors — that are not necessarily under direct control — to participate in a particular way. The way to do that, in this case, is to create a price difference over this line in such a way that the agents on both sides respond to relieve the line. The two price levels must be chosen such that (1) total demand equals total supply, and (2) the line flow is equal to its maximum. For this specific example, the prices p_{RIGHT} and p_{LEFT} indicated in Figure 4.2 (B) accomplish this. The resulting line flows are shown in Table 4.1. Note that the two conditions are met: $d_{right}(p_{right}) = -d_{left}(p_{left})$, as can be seen in the figure, and $|z_3| = |d_{left}(p_{left})| = |z_{3,MAX}|$, as is indicated in the table.

4.1.2 Related Work on Locational Pricing

Two early introductions of the concept of locational pricing for networked services were published in the beginning of the 1970s. Then, William Vickrey introduced the concept in an essay on responsive pricing of utility services, such as telephone services, road usage and energy delivery [30]. Few years later, Carson Agnew described a model for varying congestion tolls in highway and communications networks [1].

Both in computer science and in power systems economics the notion of locational pricing is used to solve a number of network-related problems. Without the aim to provide a complete overview, we will briefly discuss the usage of the concept in both fields.

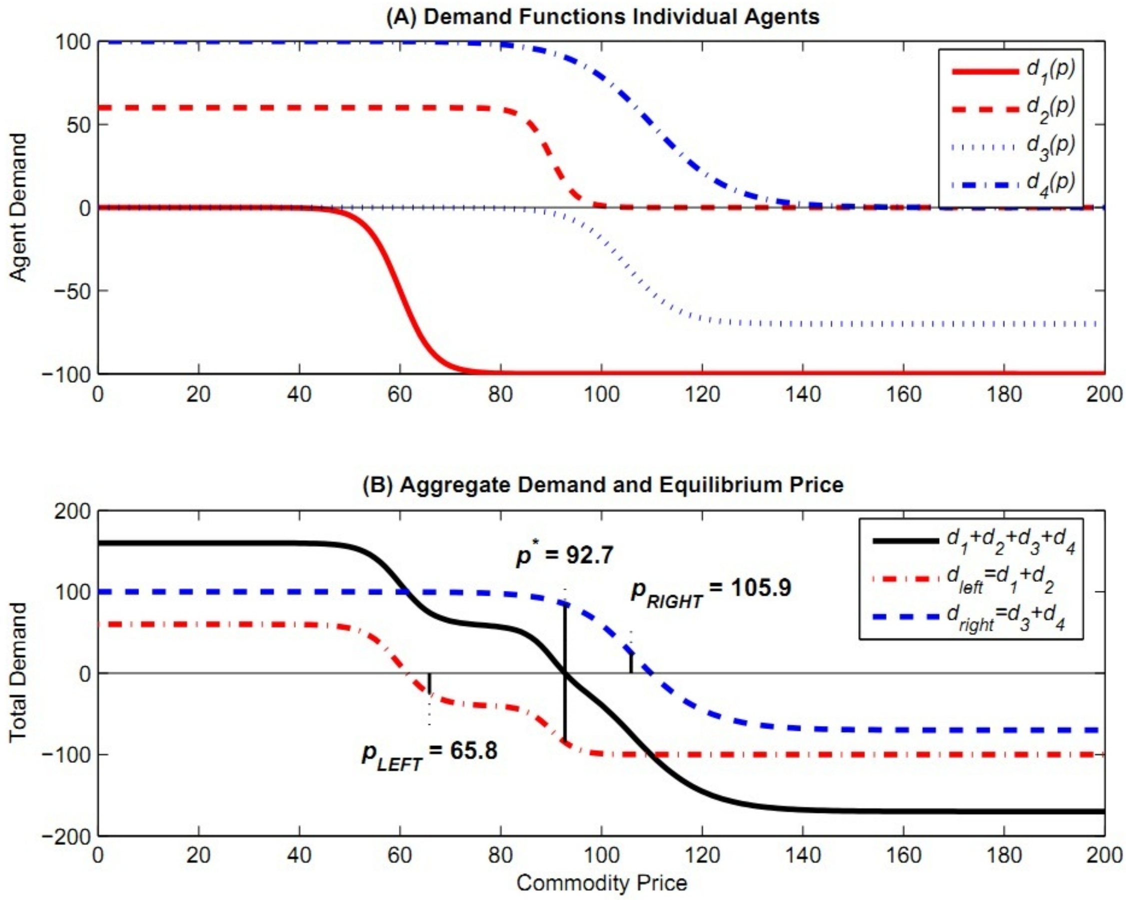

Figure 4.2: Locational prices to solve line overloading (see text).

Locational Pricing in Computer Science

Locational pricing plays an important role in different solutions to problems related to data network (e.g. internet) topologies, routing and pricing. An example of this is is work by Mullen and Wellman that describes a computational market model for information services distributed over a data network [22]. Their focus is “on the economic problem of when and where to establish mirror sites for the more popular information services. Competitive agents choose to set up mirrors based on going prices for network bandwidth, computational resources and the information service.”. Another example is work done by MacKie-Mason and Varian, who describe a basic economic theory of pricing congestible network resource such as an ftp server, a router, a Web site, etc. [20]. They examined the implications of “congestion pricing” as a way to encourage efficient use of network resources. An overview of the development of costs and pricing schemes for data infrastructure usage, against the history of that for other infrastructures, can be found in [23].

Further, the concept has been used in auction algorithms for solving the classical linear network flow problem and its various special cases such as shortest path and max-flow problems [4, 3].

Locational Pricing in Power Systems Economics

Important work on locational pricing was done within the field of electrical power systems economics. During the 1980s, Bohn et al. developed a comprehensive theory for spot pricing of electricity [5, 6]. The book *Spot Pricing of Electricity* that resulted from this research became a standard work in this field [28]. Their approach became known as *locational marginal pricing* (LMP).

Currently, LMP is increasingly applied in management of electrical power infrastructures at the level of bulk generation and transmission of electricity. For instance, in the USA states of Pennsylvania, New Jersey and Maryland (PJM), congestion problems in the high-voltage transmission network are being solved using LMP from 1998 on. In doing this, the wholesale electricity markets and transmission system power flow analysis are coupled in order to use pricing to allocate scarce transmission capacity. At times of sufficient transmission capacity, the system works as a coordinated and transparent spot market. When the transmission system is constrained, the spot prices can differ substantial across the three states [14]. Further, LMP has been implemented in a similar manner in Chile and New Zealand.

Comparable pricing mechanisms are used to optimize the utilization of transmission system interconnectors between countries in Europe. In the Nordic countries, this is known as *market splitting* as the common Nordpool wholesale market is split into two or more loosely coupled markets when interconnection capacity is constrained [19]. In the rest of Europe, individual countries are having national electricity wholesale markets since the end of the last millennium. The European transmission system operators (TSOs) and market operators are currently adopting a LMP-based market model, similar to that of the Nordic countries [11]. As, in contrast to the Nordic situation, this model aims at coupling previously separated markets, the mechanism has been coined *market splitting*.

In [13], it is shown for the UK electricity system, that moving from uniform prices to optimal locational prices could raise social welfare, lower vulnerability to market power and would also send better investment signals. On the other hand it would create politically sensitive regional gains and losses.

4.1.3 How are passive flow networks different?

As stated in the introduction of this chapter, current electronic market algorithms for flow commodities, do not take the characteristics of an underlying flow network into account. However the concept of locational pricing is used in multi-agent systems as shown in section 4.1.2. Aren't these results directly applicable to flow commodity networks? The answer lies in the type of network addressed. Real-world industrial applications of flow resource allocation use *passive* transport networks. There is no way of directing the flow to follow a particular path. Instead, the commodity flows via the path(s) of least resistance, possibly via a number of parallel trajectories, from the point of injection to the point of subtraction. In a network of a given topology, and with given resistance characteristics, the actual flows through the network depend entirely on commodity injections and subtractions at the network nodes. The flow characteristics in these networks are fundamentally distinct from those in actively switched networks, such as packet-switched data networks and road transportation networks.

4.2 Network and agent models

4.2.1 Network Model

We model a flow network by a directed graph $G = \langle V, E \rangle$, with $V = \{v_1, v_2, \ldots, v_{N_n}\}$ a set of network nodes and $E = \{e_1, e_2, \ldots, e_{N_l}\}$ a set of directed lines with associated flow characteristics. The lines are directed in order to define the positive flow direction. So, a negative flow value for a particular line indicates a flow against defined line direction. Note that this is in contradiction to directed graphs in mathematics or computer science, where it is not possible to follow a pathway against defined line directions. Each line i is defined by a tuple $e_i = (h_i, t_i, r_i, z_{i,\max})$, where:

- $h^i, t^i \in V, h^i = t^i$ are the head and tail vertices joined by the line. The positive flow direction is defined to be from head h_i to tail t_i. A negative flow value indicates a flow from tail to head.
- r_i is the resistance of e_i.
- $z_{i,\max}$ is the flow capacity of the line, the maximal allowable flow through e_i.

There is a flow model F mapping graph G to a network transfer matrix H: H $= F(G)$. This matrix holds the relation between the subtractions d_k (local demand minus local supply)[3]
at nodes $\{v_k \,/ k = [1, N_n - 1]\}$ and line flows z_i at all lines e_i:

$$z = Hd \tag{4.3}$$

The individual matrix elements H_{ik} represent the influence of the subtraction at node v_k on the flow along line e_i: $H_{ik} = \partial z_i / \partial d_k$. In other words, it gives the flow through e_i as caused by the total net demand at v_k. Accordingly, the flow along a line e_i is given by:

$$z_i = \sum_{k=1}^{N_n} \frac{\partial z_i}{\partial d_k} d_k \tag{4.4}$$

Note that, in (4.3), vector **d** holds the power demand at all nodes except node v_n. Likewise, this particular node, referred to as the *swing node*, has no corresponding column in H. This is a common property for network matrices describing a closed conservation-of-matter system in physics, as the full matrix is singular by definition. The swing node is generally indicated with a star (*). So:

$$d_{N_n} = d^* \tag{4.5}$$

For a given set of nodal subtractions $\{d^k \mid k = [1, N^n - 1]\}$, d^* follows from the conservation-of-matter property:

$$d^* = -\left[\sum_k d_k + L + \Delta S\right] \tag{4.6}$$

here L equals the total network losses and ΔS denotes the change in the amount of commodity stored in the network itself. The magnitude of L and ΔS, if they exist at all, depend on the underlying physics of the commodity in question. We will discuss both L and ΔS in greater detail later on.

4.2.2 Acyclic Networks

In the context of this text, we define an acyclic network a network without any cycles regardless the direction of the lines. In an acyclic network, there is no pathway starting at some node v_k and following a sequence of lines for each line either in the positive or negative direction that eventually leads back to v_k again. This type of network is also referred to as a *tree* or a *radial* network. We assume the swing node is the root of the tree structure, with all lines directed away from the root, i.e. for each line the head is closer to the root than the tail. Accordingly, the positive flow direction is from root towards the leaves. Further, we assume there is only power demand or supply in the leaves. This is without loss of generality as demand or supply in any non-leaf node can be modeled by a line from that node to a leaf accommodating the demand and/or supply of the tree node.
Note that, in these acyclic networks, all H_{ij} elements are either 1 or 0. This means that subtractions or injections at a certain node only influence the flow through the lines between the root of the tree and that node.

[3] Throughout this report 'demand' and 'subtractions' are defined positively. Supply can be seen as negative demand and injection as negative subtraction.

4.2.3 Agent model

There is a set of agents $X = \{x_1, x_2, \ldots, x_{N_n}\}$, each agent x_k representing the demand and/or supply at node v_k . The agent holds a demand function $d(p)$ stating the agent's demand against resource price p. Each agent must act as a rational trader, i.e. its demand function $d_k(p)$ is continuous and monotonically decreasing.

4.2.4 Network-agnostic market clearance

The set of agent demand functions define an allocation problem. As the set of agents represents a closed system, the problem is finding an allocation of electrical power for each agent that balances demand and supply. When ignoring the network, the allocation problem is solved by finding the general equilibrium price p^* such that:

$$\sum_{k=1}^{N_n} d_k(p^*) = 0 \tag{4.7}$$

Under these conditions market clearance is established, i.e. total demand equals total supply in the agent set. According to general equilibrium theory in microeconomics, the general equilibrium solution is *Pareto* optimal, a social optimal outcome for which no other outcome exists that makes one agent better-off without making other agents worse-off [21]. From a computational point of view, electronic equilibrium markets are distributed search algorithms aimed at finding the best trade-offs in a multidimensional search space defined by the preferences of all agents participating in the market [32].
When using this network-agnostic solution to the allocation problem, one implicitly assumes the network has virtually infinite capacity and that network losses are negligible.

4.2.5 Flow Model

A complete discussion of network flow analysis methodologies is beyond the scope of this document. Hence, we briefly discuss a general steady-state flow model here. Congestion management in flow infrastructures is a process that takes place on a time scale of minutes, so, a steady-state model is sufficient for this purpose. The model described here is used later on to indicate the role of the flow model in the market algorithms described.
The flow model F maps the graph G to a network transfer matrix: H = F (G). Here, the two important properties of G are its topology and the resistance values r_i of the graph's edges $e_i \in E$. Incidence matrix A with size $(N - 1) \times N_l$, represents the graph topology and is defined as:

$$R_{ij} = \begin{cases} r_i, & \text{if } i = j \\ 0, & \text{Otherwise} \end{cases} \tag{4.8}$$

$$\quad 0,\ \text{Otherwise}$$

Resistance matrix R with size $N_l \times N_l$, is defined as:

(4.9)

From these two matrices the transfer matrix can be calculated as follows:

$$B = (A^T R^{-1} A)^{-1} \tag{4.10}$$
$$H = R^{-1} AB \tag{4.11}$$

Each column k of H describes the flow path of the commodity subtracted at node v_k from the swing node. The flow values resulting from equation (4.3) can be regarded as a superposition of all nodal demands being transported from the swing node. Figures 4.3 and 4.4 illustrate this for a triangular network, composed of three nodes and three lines. The former figure shows the corresponding network transfer matrix. The latter visualizes the superposition of flows caused by individual nodal subtractions. The left side of the figure shows the resulting flows for subtractions $d = (-1, 1)^T$, in a network model without losses and inherent storage. The resulting flows can be regarded as a superposition of two effects, as shown in the right side of the figure. Firstly, injecting commodity at node 1 and the resulting flow to the swing node, and secondly, a subtraction at node 2, resulting in a flow from the swing node.

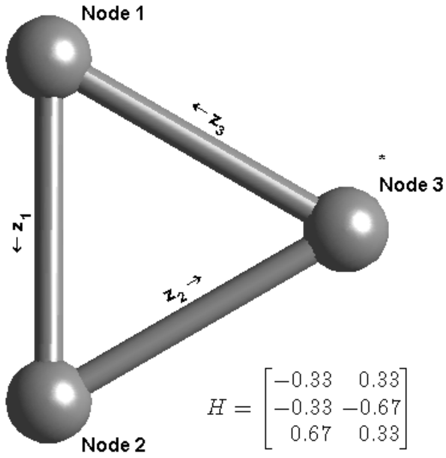

Figure 4.3: A triangular network with corresponding transfer matrix H . The positive line directions are shown by the arrows in the line labels. Node 3 is chosen to be the swing node and the resistances of the individual lines are chosen to be equal.

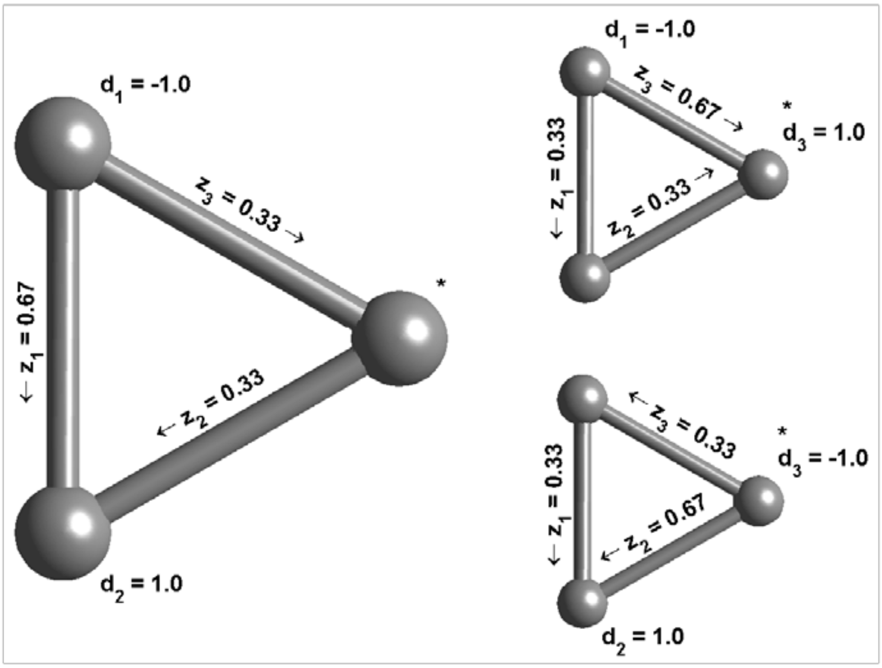

Figure 4.4: Network flows as a result of demands $d_1 = -1$ and $d_2 = 1$ (right). Superpositional decomposition of these flows into $d=(-1, 0)^T$ (top left), and $d=(0, 1)^T$ (bottom left).

4.3 Locational Marginal Pricing Framework

Equation 4.7 in section 4.2.4 gave the general equilibrium equation for an network-agnostic market clearing process. Now, the challenge is to generalize this market into one that does take the characteristics of the underlying network into account. These network characteristics may have three aspects relevant to this process. Firstly, the commodity flow zi through each line i will have a maximum allowable flow $z_{i,\mathrm{MAX}}$. Secondly, the total network flow may — depending on the physics of the commodity type— induce network losses *L*, and, thirdly, the amount of commodity stored in the network itself may —again, commodity type dependent— change at a certain time. Changes in *inherent storage* are denoted by ΔS.
When these network characteristics are accounted for, the market-based optimization in (4.7) generalizes to finding a set of locational prices ($p_1, p_2, \ldots p_{N_n}$), one for each network node k, such that:

$$\sum_{k=1}^{N_n} d_k(p_k) + L + \Delta S = 0 \tag{4.12}$$

$$|z_i| \leq z_{i,\mathrm{MAX}}, \forall i \tag{4.13}$$

The first equation ensures market clearing: the total demand and supply in the network, plus the total network losses L, plus the change in inherent network storage ΔS must equal to zero. The set of inequalities (4.13) gives a line capacity constraint for each individual line in the network. Note that these equations reduce to the general equilibrium equation (4.7) when transport losses and inherent storage are absent and all line capacities are sufficient. In that case, all nodal prices p_k become equal to the general equilibrium price p^*.
A decomposition of the locational prices p_k into specific components completes the framework. Each component enforces the market outcome to obey one aspect in the equations (4.12) and (4.13). The price decomposition is defined by:

$$\begin{aligned} p_k(t) = {} & \lambda(t) && \text{[Market Clearing Component]} \\ & + \eta_{C,k}(t) && \text{[Line Capacity Component]} \\ & + \eta_{L,k}(t) && \text{[Network Losses Component]} \\ & + \eta_S(t) && \text{[Network Storage Component]} \end{aligned} \tag{4.14}$$

Note that, in general, the demand functions of individual agents are changing over time. Hence, the time-dependency of the market price and its components in (4.14). However, we omit the time dependency from here on for reasons of readability.
Using this pricing scheme, the locational prices in a flow network at a certain time depend on:

- **Demand & Supply**: The total demand and supply in the network, subject to the preferences of all individual agents.
- **Network**: The availability of flow capacity, plus, depending on the physical characteristics of the flow commodity, network losses and/or changes in the amount of commodity stored in the network itself.
- **Spatial demand/supply distribution**: The specific locations of production and consumption in the network.

In the next subsections we will discuss the four price components in detail.

4.3.1 Market Clearing Component

The Market Clearing Component λ is the commodity price component used to balance the total demand and supply in the system. This component is equal for all agents attached to the network; there is no locational aspect in this component. In the absence of line capacity constraints, losses and inherent storage, λ is equivalent to the general equilibrium price.

4.3.2 Line Capacity Component

The Line Capacity Component $\eta_{C,k}$ is a price mechanism to allocate the use of scarce network capacity [28]. The component becomes large in magnitude when the maximum capacity of network lines is being approached. This is one of the price components that brings locationality into the pricing scheme: the magnitude of $\eta_{C,k}$ is dependent on the location in the network and can be different for each network node k. Each $\eta_{C,k}$ is chosen in such a way that all network flow magnitudes are less than or equal to the maximum capacity of the individual network connections, obeying (4.13).

Assume in a network one line, line i with flow z_i, is overloading. We treat that line's transport capacity as a scarce resource and let a price mechanism allocate its use among the agents. In this case of the single overloaded line i, $\eta_{C,k}$ is given by:

$$\eta_{C,k} = \theta_i \frac{\partial z_i}{\partial d_k} \tag{4.15}$$

Thus, the price component at network node k resulting from this overloaded line i is equal to some term θ_i multiplied by the incremental flow through i as caused at node k. A market clearing mechanism is used to find an appropriate value for θ_i. So, during price forming, θ_i is adjusted until consuming and producing agents respond by changing their usage or production so that the line overload does not occur.

Equation (4.15) gives $\eta_{C,k}$ for the situation where only line i overloads. The full equation becomes:

$$\eta_{C,k} = \sum_{i=1}^{N_l} \theta_i \frac{\partial z_i}{\partial d_k} \tag{4.16}$$

where N_l is the number of lines in the network. Naturally, θ_i needs to be nonzero, only if line i would be overloaded otherwise. Hence the condition:

$$|z_i| \leq z_{i,MAX} \leftrightarrow \theta_i = 0, \ \forall i \tag{4.17}$$

Dependent on the sign of the partial derivative term, $\eta_{C,k}$ can be positive or negative.

Those locations k, where an increase in demand d_k leads to a decrease in flow z_i, have a negative $\eta_{C,k}$. Since both producers and consumers at k are having the same locational price p_k, and increasing demand at k will have an equal effect as decreasing supply at k, both producers and consumers have equal incentive to respond to prevent the line overload. Further, note that actors having a higher 'network distance' from an overloaded line will have a lower influence on the flow over that line. Hence, their $\partial z_i/\partial d_k$ is lower and, accordingly, their incentive to respond is lower.

4.3.3 Transport Network Losses

One of the special phenomena occurring when flow commodities are transported is the loss of commodity. A resource allocation method must take these losses into account. If not, the theoretical commodity balance found by the method will yield commodity imbalance in practice. Naturally, the losses add to the total demand in the network. For physical flows in the gas or liquid phase these losses are pressure-driven (e.g. filtration or permeation of pipes and connections). For electrical energy these losses are Ohmic.

Losses can be incorporated in the market search by considering the associated costs, which are equal to the loss magnitude L times the commodity price λ. As different network locations will face different losses, different network nodes k may have different magnitudes for $\eta_{L,k}$. The marginal cost for network losses at location k is given by [28]:

$$\eta_{L,k} = \lambda \frac{\partial L}{\partial d_k} \tag{4.18}$$

where L equals the total network losses at time t. Thus, the network losses price component at node k is equal to the commodity price multiplied by the incremental system losses as caused at k. When the losses of an individual line depend on the actual line flow, then $L = \sum_i L_i\ [z_i\ ]$, and (4.18) can be expanded into:

$$\eta_{L,k} = \lambda \sum_{i=1}^{N_l} \frac{\partial L_i[z_i]}{\partial z_i} \frac{\partial z_i}{\partial d_k} \tag{4.19}$$

4.3.4 Network Inherent Storage

For specific flow commodities there is a certain amount of the commodity contained in the flow network. The stored amount may change over time, when the total feed-in to the network is unequal to the total feed-out. Generally, inherent storage is possible in case of gas flows, where it is influenced by the average system pressure. The magnitude of ΔS is the result of the spatial distribution of demand and supply in the network.
Price component η_S is defined as:

$$\eta_S = \lambda\ \Delta S \tag{4.20}$$

Since ΔS can't be accounted to specific nodes, as is the case with network losses, the cost (or benefit) of the storage changes are accounted for regardless of locationality.

4.3.5 The Locational Price

Substituting the above results in (4.14) yields :

$$\begin{aligned} p_k &= \lambda + \sum_{i=1}^{N_l} \theta_i \frac{\partial z_i}{\partial d_k} + \lambda \frac{\partial L}{\partial d_k} + \lambda\ \Delta S \\ &= \lambda \left(1 + \frac{\partial L}{\partial d_k} + \Delta S\right) + \sum_{i=1}^{N_l} \theta_i \frac{\partial z_i}{\partial d_k} \end{aligned} \tag{4.21}$$

4.4 Analysis

4.4.1 Search Space and Convergence

In market-based resource allocation, each agent's demand function $d_a(p)$ is generally required to be continuous and monotonically decreasing. A general equilibrium search (4.7) tries to find a root of the aggregate demand function $\sum_a d_a(p)$, which is also a continuous, monotonically decreasing function. Thus, if a solution exists (i.e. there is sufficient elasticity in supply and demand), this is a unique equilibrium point and the search is guaranteed to converge to it.
In our case, price forming is a search in a space of (N_l + 1) dimensions. This search space is defined by (λ, θ_1, θ_2, , θ_{N_l}). Any set of values for these parameters yields a set of locational prices (p_1 , p_2 , . . . p_{N_n}) according to equation (4.21). These prices must be chosen such that the market clears (4.12) and the line capacity constraints (4.13) are met.

The λ price component determines the supply/demand balance in the network. By substituting the locational price equation (4.21) in the commodity balance constraint (4.12) while omitting L and ΔS, the search space along the λ-dimension is obtained:

$$\sum_{k=1}^{N_k} d_k \left(\lambda + \sum_{i=1}^{N_l} \theta_i \frac{\partial z_i}{\partial d_k} \right) = 0 \tag{4.22}$$

Since all d_k are continuous and monotonically decreasing, the left-hand side of this equation shares these properties. Consequently, if for a given set of θ_i values there exists a λ such that total demand equals total supply, this solution is unique and a search in λ will converge to it.

Each individual θi ensures the line capacity constraint of one line is met. To assess its convergence properties, suppose we vary θ_i while λ and all other θ-values remain stationary. Then, for all nodes k with $H_{i,k} = \partial z_i / \partial d_k = 0$, an increase in θ_i will result in a change in nodal demand in a direction opposite

$$\frac{\partial z_i}{\partial d_k} > 0 \Rightarrow p_k \uparrow_{\theta_i} \Rightarrow d_k(p_k) \downarrow_{\theta_i}$$

$$\frac{\partial z_i}{\partial d_k} < 0 \Rightarrow p_k \downarrow_{\theta_i} \Rightarrow d_k(p_k) \uparrow_{\theta_i}$$

to the sign of $H_{i,k}$. In short:

where $\uparrow_{\theta_i}$ denotes “continuously and monotonically increasing in θ_i ”. The first step follows from the nodal price definition (4.21), the second from the requirement of demand functions to be defined as continuously and monotonically decreasing.

Both $H_{i,k}$ and $d_k(p_k)$ influence flow z_i, according to (4.4), such that:

$$\left.\begin{array}{l} \frac{\partial z_i}{\partial d_k} > 0 \\ d_k(p_k) \downarrow_{\theta_i} \end{array}\right\} \Rightarrow \quad z_i \downarrow_{\theta_i}$$

$$\left.\begin{array}{l} \frac{\partial z_i}{\partial d_k} < 0 \\ d_k(p_k) \uparrow_{\theta_i} \end{array}\right\} \Rightarrow \quad z_i \downarrow_{\theta_i}$$

Thus, for every node with a nonzero influence on z_i , an increase in θ_i will result in a decrease in z_i . Consequently, if any line i is overloaded, there is a unique value for θ_i where $z_i = z_{i,max}$. As z_i is continuously and monotonically decreasing in θ_i, a search will converge to this solution, provided there is enough elasticity in those demands $d_k(p_k)$ for which $H_{i,k} = 0$.

4.4.2 Combining Locational Pricing and Flow Analysis

Due to the swing node’s absence in the network transfer matrix H , some special features arise that are important when solving the optimization problem:

Market Clearing

The demand of the swing node can be computed in two distinct ways, denoted here as d^{*1} and d^{*2}, respectively. The first one follows from the flow analysis. Taking (4.6), and assuming the swing node has the highest node number, yields:

$$d^{*1} = -\left[\sum_{k=1}^{N_n - 1} d_k(p_k) + L + \Delta S \right] \tag{4.23}$$

Secondly, the swing node demand follows from the local demand function and the local price:

$$d^{*2} = d_k(p_k), \text{ with } k = N_n \tag{4.24}$$

It may be clear that $d^{*1} = d^{*2}$ must hold in a sound solution to the optimization problem. Moreover, any set of prices p_k , $k = 1 \ldots N_n$ that results in equal values for d^{*1} and d^{*2} complies with the commodity balance constraint (4.12).

Swing Nodal Price

Being the balancing item in the flow calculation, demand or supply at the swing node has no modelled influence on any flow in the network. In other words, the flow model assumes that:

$$\frac{\partial z_i}{\partial d^*} = 0, \ \forall i \tag{4.25}$$

Substituting above equation in (4.21) and using the expanded loss component (4.19) yields the swing node price:

$$p^* = \lambda \, (1 + \Delta S) \tag{4.26}$$

So, while the demand or supply at the swing node has no effect on the network flows, its local price has no effect on line congestions and line losses. The price at the swing node only depends on those price components that have no associated locational aspects.
In case the network exhibits no or negligible inherent storage, as is the case with electricity, the swing node price becomes:

$$p^* = \lambda \tag{4.27}$$

Losses and Line Capacity Components

As described in section 4.2.5, the result of a load flow calculation is a superposition of all nodal demands being transported from the swing node. The H -matrix describes the flow paths from the swing node to every individual node. As a result, for a node k, all $H_{i,k} = \partial z_i / \partial d_k$ values for lines i that are not part of a possible flow path between node k and the swing node are equal to zero. As a consequence, the losses component $\eta_{L,k}$ for any node k is the price for the losses of transporting the demand at k from the swing node. Similarly, the line capacity price component $\eta_{C,k}$ is only influenced by the lines that are in a possible flow path between k and the swing node.

4.5 Example

Consider the PowerMatcher coordinated island network in Figure 4.5, which contains a swing generator S, an industrial load R_I and two identical districts A and B, represented by the buses B_5 and B_6 respectively. Each district contains a load (R_A and R_B) and a generator (G_A and G_B). The districts are connected to a common bus B_2 by lines e_4 and e_5. Bus B_2 is connected to bus B_1 by a long line e_1 . It is assumed that all loads and generators in the network have flexibility to some extent. In this example, only active power is being considered, thus no voltage levels and reactive power are taken into account. Each load and generator is represented by a device agent that buys or sells electricity against the marginal costs of the load or generator. The auctioneer of this network resides at bus B_1 and there are concentrators at buses B_2, B_5 and B_6. There is no objective agent, because the network is operated as an island. Four cases have been considered and the results are shown in table 4.3[4]. Additionally, the demand functions of the individual agents the transformed and non-transformed aggregated demand functions of the concentrators in the four cases have been plotted in Figure 4.6. To

[4] In this example the units of power and price have been left out on purpose to increase the readability

provide a reference, the example network was first calculated without any constraints and thus, the market clearing price of 4.75 was the same for the global and local markets.

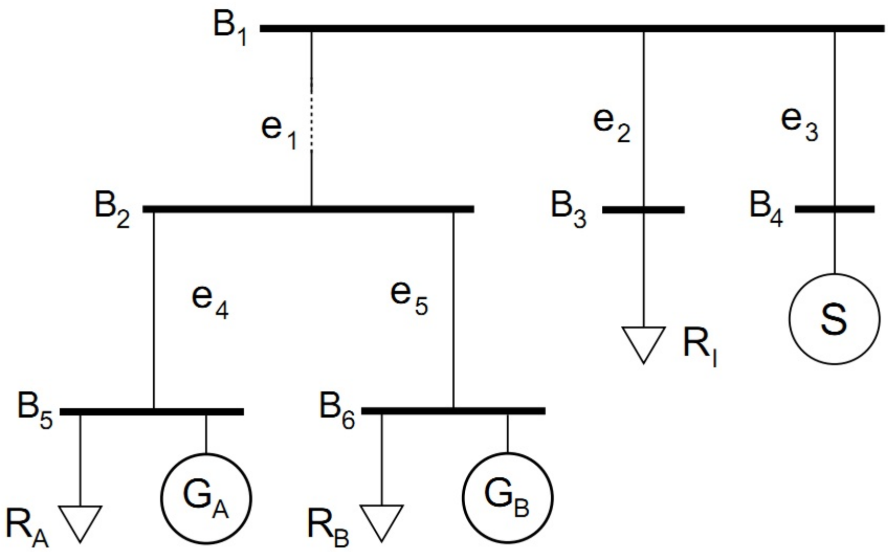

Figure 4.5: One-line representation of the example network.

The second case considered a capacity flow limit of the line e_4, which was implemented by using the propagation operator as defined in (4.28). The maximum capacity for this line was set to $z_{5,\max} = 15$. As a result, the global market price has decreased with respect to the reference case, while the local market price in district *A* has increased. In the reference case the load on line e_4 was 24.06, which violates the maximum capacity that exists for that line in this case. With the price increase, the production has increased and the demand decreased, such that the load on the line is exactly 15. Consequently, district *A* demands less electricity on the global market, making the price in the unconstrained districts to go down.

Table 4.3: Demand allocation and price at given nodes and buses for four cases.

		Case 1	Case 2	Case 3	Case 4
Demand	R_A	29.85	28.86	28.26	28.86
	G_A	–5.78	–13.86	–16.21	–13.86
	R_B	29.85	30.00	28.26	29.59
	G_B	–5.78	–0.39	–16.21	–8.62
	R_I	1.71	3.61	1.44	2.63
	S	–49.84	–48.21	–49.95	–49.19
Price	B_1	5.76	5.20	5.86	5.46
	B_2	5.76	5.20	6.38	5.93
	B_5	5.76	6.22	6.38	6.22

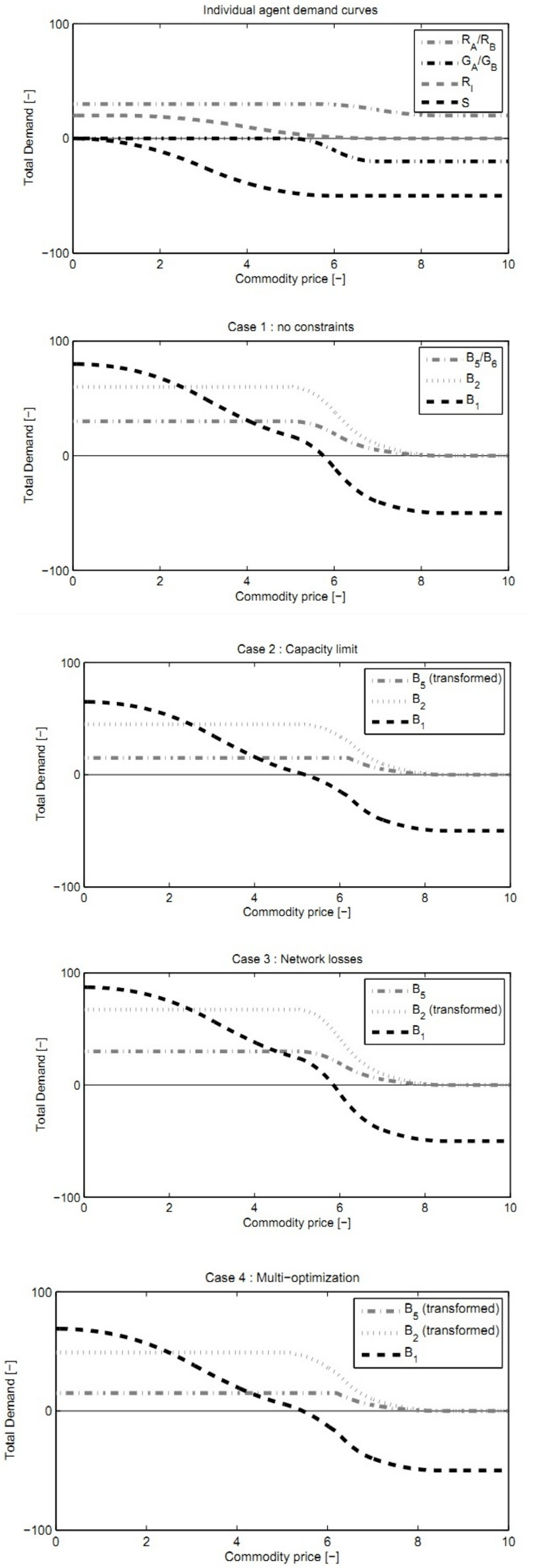

Figure 4.6: Demand curves of the individual loads and generators and the aggregated demand curves for different network nodes in the four cases that were considered.

The third case considered a significant energy losses on line e_1 with respect towards the reference case, i.e. there were no other constraints in the network, using the operator in (4.29), with $l_{1c} = 2 \cdot 10^{-3}$. The losses in line e_1 were 24.41, which were mainly compensated by an increase of production of generators G_A and G_B. A positive side effect of this, is that the net demand and load on line e_1 and thus, the losses in this line are lower of what they could have been if no locational pricing would have been applied.

In the last case, the capacity limit on line e_4 and the losses in line e_1 were introduced simultaneously in the example network. Consequently , the concentrator at B_2 propagates an already propagated demand function, thus including the optimization of the capacity limit within the optimization of the line losses. The results are not surprising. In district *A*, the capacity limit is dominant and also affected the amount of line losses in e_1, which were 10.59.

4.6 Conclusion

Current methods for market-based allocation of flow resources ignore transport network characteristics and constraints. This limits their applicability in larger-scale industrial applications, which often are distributed over a large regional area and use congested transport networks. In this chapter we have presented a theoretical framework, based on a framework in power systems economics, and an algorithmic method for finding transport network feasible solutions in market-based flow resource allocation. The framework describes a pricing scheme that enforces the electronic equilibrium market to find solutions that are feasible for the underlying transport network, i.e. obeying network constraints and accounting non-constraining network characteristics such as network losses and network-inherent storage. This pricing scheme is generally applicable to all types of flow resources. The constrained optimization problem that follows from the theoretical framework is solved by the distributed algorithm described in the second part of the paper. We have shown that, under the common condition of demand and supply elasticity, this algorithm converges to a unique solution. Further, we have demonstrated the algorithm for a medium-sized example network.

5. Discussion

In the previous sections, we argued about the necessity of introducing distributed control in the electricity infrastructure in order to cope with the interrelated trends of increasing sustainable electricity sources and distributed generation. We have shown how a specific implementation of distributed control can be used for commercial portfolio balancing as well as for DSO congestion management. An important remaining question is: how to combine the two?
Such a dual-objective coordination mechanism needs to be designed for a future electricity system characterized by:

- Distributed Generation and Demand Response are a substantial factor in the electricity markets.
- A substantial portion of central generation is off-shore wind.
- Market parties and network operators optimize their stakes using the DER in their portfolio, or in their network area, respectively. Dependent on the situation, these stakes may be conflicting at one time and non-conflicting at another time.
- Incentives to market parties (generators, suppliers, and end users alike) reflect the true costs of both generation and infrastructure. On the one hand, this will increase efficient usage of the infrastructure (network load factor optimization) and on the other hand it gives the right market signals for investment decisions (Generation against Demand Response against Infrastructural investments).

Figure 5.1 shows an architecture that supports the market situation described above. It is a setting with multiple Market Parties (Balancing Responsible Parties, BRPs), each running a commercial virtual power plant (CVPP), and multiple Distribution System Operators (DSOs), each running a technical virtual Power Plant (TVPP). In the Figure the CVPPs are represented by the blocks labeled ”Commercial Aggregation” and the TVPPs by those labeled ”Network Service Aggregation”.
A BRP has special interests:

- Desire to aggregate a high number of DER units, as this smoothens-out the stochastic behavior of the individual DER.
- Aspiration to spread its DER portfolio over a big (national) area to increase spatial smoothing of weather influences on DG and on responsive loads.
- Has no locational aspects attached to the desired portfolio behavior for most of its operational parameters.
- Avoids balancing costs when their portfolio as a whole is in balance.

As a result the commercial portfolio of a BRP is most likely located in the grid area of more than one DSO.

A DSO has special interests as well:

- Preference to address only the DER units in its grid area, sometimes even dependent on individual grid cells or segments.
- Desire to incentivise DER to deliver system management services.
- Has a locational aspect in the desired behavior or the DER in its network.
- Avoids investments in infrastructural components by active management of the DER in their network.

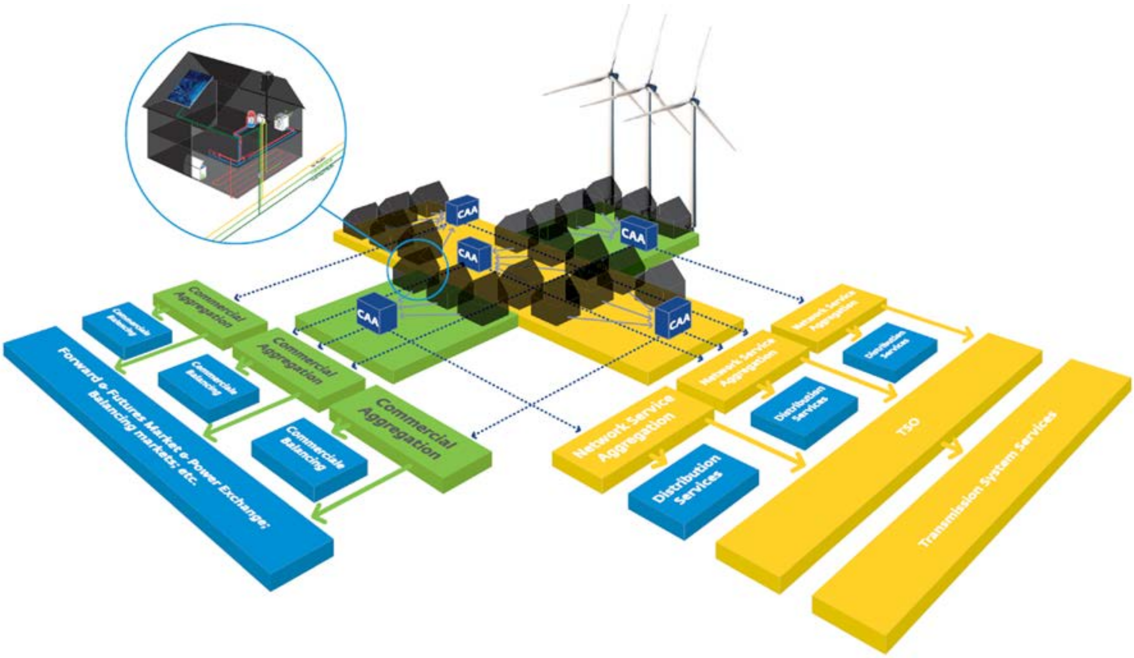

Figure 5.1: Orthogonal dual market-based architecture for commercial and technical VPPs in future electricity systems.

In the orthogonal dual architecture, one CVPP has to deal with several TVPPs and one TVPP with several CVPPs. The individual DER units at the premises of one customer, in the Figure represented by a house, communicate with CVPP components only. The Commercial Aggregating Agent (CAA) aggregates all DERs in the portfolio of the corresponding BRP located in a common grid area. Each CAA provides commercial services directly to its CVPP, but it also provides local grid services to the DSO. Thus, each CAA responds to incentives of both the CVPP it is part of and the TVPP that covers its grid area. The stakes of BRP and DSO come together at this point. When these stakes are non-counteracting, the CAA can deliver the services requested by the DSO for a lower price compared to the situation in which the stakes do counteract. Accordingly, those CAAs without an internal conflict will respond to both the CVPP and TVPP request first. In this way, flexibility services from DER will be used based on merit order and the stakes of the different parties will be balanced automatically against each other.

6. Conclusions and recommendations

6.1 Conclusions

The increase in distributed generation requires the gradual introduction of a new control philosophy in the power market and infrastructure. A shift needs to be made from the control of a few large centralized generators towards the control of a large number of small Distributed Energy Resources. These distributed energy resources include distributed generators, demand response and distributed energy storage. Their operation is very dependent upon momentary local circumstances, such as the availability of renewable energy supply, the customer demand and the buffer levels of energy buffers. Centralised control of Centralized Generation needs to change towards decentralized control of Distributed Energy Resources.

Multi Agent Systems (MAS), such as PowerMatcher, can provide for this decentralized control. They are scalable, flexible and open for extensions. They can at the same time support market functions such as balancing islanded system and grid functions such as avoiding network constraints (using locational marginal pricing).

A decision / coordination system for PowerMatcher has been modeled in algorithms and implemented in a simulation tool. The simulations show the feasibility of the algorithms. In the case simulated PowerMatcher coordination resulted in a 40% lower total production (energy) by the the diesel generator, resulting in 40% less energy use and emissions. The peak load served by the diesel generator was even reduced by 45%, resulting in substantially lower investments for the diesel generator and the supporting distribution grid.

To achieve the goals of all actors involved, it is proposed to introduce a dual objective architecture in for the PowerMatcher. This can reconcile the objectives of the participants in the energy market with those of active network management.

6.2 Recommendations

6.2.1 Recommendations for the EIT project

This report describes the results of the ECN contribution to theme C of the EIT project. TU/e is still completing its contribution. The following recommendations for that work follow from the work described above:

- In the literature study (WP12) attention should be paid to the failure times allowed by the network and by the market. How can the coordination system stay within these limits? What are the resulting requirements for the ICT architecture and the network devices?

- The review of the existing situation (WP13) should focus on the existing systems, requirements and actors. How can the coordination system be intregrated in this situation? What are the additional requirements that must be met?

- The development of the information model (WP15) should focus on priority issues. Which processes should be prioritized?

- The verification task (WP17) should include a simulation to verify the information model.

- In the dissemination task (WP19) as well as project theme D on Market Perspectives issues such as integration with existing systems and business processes should be checked as well as timing issues.

6.2.2 General recommendations

Several other recommendations for further research - outside the scope of the EIT project - can be made.

- Develop a symbiosis of ICT architecture and network devices. In such a layout the network devices can work semi-autonomously (without communication) to support the power system in case the ICT network fails.

- Develop a dual objective architecture in which network and market objectives can be reconciled.

- Develop protection mechanisms for smart grid, supporting islanding, reverse energy flows etc. This is an extensive work field, which is essential if a smart grid is to meet the requirements for safety and availability of the traditional power grid.

- Check the applicability of the proposed coordination mechanism at the level of transmission grids. Many of the tools and architectures developed for the control of distribution grids with distributed generation may also be valuable at the level of transmission grids, especially now that transmission grids and power markets of various European member states are increasingly linked.

Bibliography

[1] Carson E. Agnew. The theory of congestion tolls. *Journal of Regional Science*, 17(3):381–393, 1973.

[2] Hans Akkermans, Jos Schreinemakers, and Koen Kok. Microeconomic distributed control: Theory and application of multi-agent electronic markets. In *Proceedings of CRIS 2004 - 2nd International Conference on Critical Infrastructures*, 2004.

[3] D. Bertsekas. An auction algorithm for the max-flow problem. *Journal of Optimization Theory and Applications*, 87(1):69–101, October 1995.

[4] Dimitri P. Bertsekas. Auction algorithms for network flow problems: A tutorial introduction. *Computational Optimization and Applications*, 1:7–66, 1992.

[5] Roger E. Bohn. Spot Pricing of Public Utility Services. PhD thesis, MIT, 1982. Also available as MIT Energy Lab Technical Report MIT-EL 82-031.

[6] Roger E. Bohn, Michael C. Caramanis, and Fred C. Schweppe. Optimal pricing in electrical networks over space and time. *Rand Journal of Economics*, 15(3):360–376, Autumn 1984.

[7] Rajdeep K. Dash, David C. Parkes, and Nicholas R. Jennings. Computational mechanism design: A call to arms. *IEEE Intelligent Systems*, 18(6):40–47, November/December 2003.

[8] Michael ten Donkelaar and Martin J.J. Scheepers. *A socio-economic analysis of technical solutions and practices for the integration of distributed generation*. Technical Report ECN-C–04-011, ECN, 2004.

[9] Energy Information Administration. *www.eia.doe.gov/emeu/international/ RecentElectricityGenerationByType.xls*, December 2008.

[10] ENIRDGnet. *Concepts and opportunities of distributed generation: The driving European forces and trends.* Project Deliverable D3, ENIRDGnet, 2003.

[11] ETSO and EuroPEX. *Flow-based market coupling; a joint ETSO–EuroPEX proposal for cross-border congestion management and integration of electricity markets in Europe. Interim report*, September 2004.

[12] Nicolai Feilberg et al. Basis for demand response. Efflocom Project Deliverable 1, Sintef Energy Research, november 2003.

[13] Richard J. Green. Electricity transmission pricing: How much does it cost to get it wrong? Cambridge Working Papers in Economics 0466, Faculty of Economics, University of Cambridge, November 2004

[14] W. Hogan. Getting the prices right in Pennsylvania–New Jersey–Maryland (PJM). Electricity Journal, 11:61–67, August/September 1998.

[15] International Energy Agency. *Distributed Generation in Liberalised Electricity Markets.* International Energy Agency, Paris, France, 2002.

[16] International Energy Agency. *The Power to Choose – Demand Response in Liberalized Electricity Markets*. International Energy Agency, Paris, France, 2003.

[17] Koen Kok, Zsofia Derzsi, Jaap Gordijn, Maarten Hommelberg, Cor Warmer, Rene Kamphuis,

and Hans Akkermans. Agent-based electricity balancing with distributed energy resources, a multi-perspective case study. In Ralph H. Sprague, editor, *Proceedings of the 41st Annual Hawaii International Conference on System Sciences*, page 173, Los Alamitos, CA, USA, 2008. IEEE Computer Society.

[18] Koen Kok, Cor Warmer, and René Kamphuis. PowerMatcher: multi-agent control in the electricity infrastructure. In *AAMAS '05: Proceedings of the 4th int. joint conf. on Autonomous Agents and Multi-agent Systems*, volume industry track, pages 75–82, New York, NY, USA, 2005. ACM Press.

[19] Tarjei Kristiansen. Congestion management, transmission pricing and area price hedging in the nordic region. *International Journal of Electrical Power & Energy Systems*, 26(9):685–695, November 2004.

[20] Jeffrey K. MacKie-Mason and Hal R. Varian. Pricing congestible network resources. *IEEE Journal on Selected Areas of Communications*, 13(7):1141–1149, 1995.

[21] A. Mas-Colell, M. Whinston, and J. R. Green. *Microeconomic Theory*. Oxford University Press, 1995.

[22] Tracy Mullen and Michael P. Wellman. A simple computational market for network information services. In *Proceedings of the First International Conference on Mutiagent Systems ICMAS-95*, 1995.

[23] Andrew M. Odlyzko. The evolution of price discrimination in transportation and its implications for the internet. *Review of Network Economics*, 3(3):323–346, September 2004.

[24] Angelika Pullen, Liming Qiao, and Steve Sawyer (eds). Global wind 2008 report. Market report, Global Wind Energy Council, March 2009.

[25] Bart Roossien. Field-test upscaling of multi-agent coordination in the electricity grid. In *Proceedings of the 20th International Conference on Electricity Distribution CIRED*. IET-CIRED, 2009.

[26] Bart Roossien, Maarten Hommelberg, Cor Warmer, Koen Kok, and Jan Willem Turkstra. Virtual power plant field experiment using 10 micro-CHP units at consumer premises. In *SmartGrids for Distribution, CIRED Seminar*, number 86. IET-CIRED, 2008.

[27] Tuomas W. Sandholm. Distributed rational decision making. In Gerhard Weiss, editor, *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, pages 201–258. The MIT Press, Cambridge, MA, USA, 1999.

[28] Fred C. Schweppe, Michael C. Caramanis, Richard D. Tabors, and Roger E. Bohn. *Spot Pricing of Electicity*. Kluwer Academic Publishers, 1988.

[29] Christof Timpe. Smart-a project. Presentation, Oko Institut, August 2009.

[30] William Vickrey. Responsive pricing of public utility services. Bell Journal of Economics and Management Science, 2(1):337–346, Spring 1971.

[31] Cor Warmer, Maarten Hommelberg, Bart Roossien, Koen Kok, and Jan Willem Turkstra. A field test using agents for coordination of residential micro-chp. In *Proceedings of the 14th Int. Conf. on Intelligent System Applications to Power Systems (ISAP)*. IEEE, 2007.

[32] Michael P. Wellman. A market-oriented programming environment and its application to distributed multicommodity flow problems. *Journal of Artificial Intelligence Research*, 1:1–23,

1993.

[33] Christoph Wolfsegger, Marie Latour, and Michael Annett. *Global market outlook for photovoltaics until 2012 – facing a sunny future*. Market report, European Photovoltaic Industry Association, February 2008.

[34] Fredrik Ygge. *Market-Oriented Programming and its Application to Power Load Management*. PhD thesis, Department of Computer Science, Lund University, Sweden, 1998. ISBN 91-628-3055-4.

[35] Fredrik Ygge and Hans Akkermans. Resource-oriented multicommodity market algorithms. *Autonomous Agents and Multi-Agent Systems*, 3(1):53–71, 2000. Special Issue Best Papers of ICMAS–98.

Appendix A. Market Algorithm for locational marginal pricing

In this appendix we describe an agent-based market algorithm for locational marginal pricing LMP). The algorithm solves the constrained optimization problem described in chapter 4. In chapter 4 we derived a framework for market based coordination in passive flow networks under network capacity constraints. The algorithm described here is an implementation of this framework. For reasons of clarity - but without loss of generality - we omit the price components for losses and inherent storage in the descriptions. Both characteristics can be added easily by implementing models for L and ΔS and incorporating these in the code lines where the nodal prices are calculated.

A.1 Algorithm Description

The algorithm is distributed over three types of agents: an Auctioneer Agent, a Node Agent for every node and a Line Agent for every line. The Auctioneer is responsible for concerting the optimization process of each market round and for searching for the λ value that clears the market, i.e. minimizes the difference between equations (4.23) and (4.24). The individual Line Agents determine their own θ_i value in order to solve capacity constraint violations, if any. The Node Agents communicate their preferences for consumption or production of the commodity at the start of each market run. Afterwards, they receive their nodal price and implement their allocation. We assume the presence of only one consuming and/or producing agent per node. When more agents are present at one node the Node Agent becomes an aggregator of all connected agent's preferences. Below we give pseudocode for all three agent types.

The pseudocode of the AUCTIONEERAGENT is given by:

```
AUCTIONEERAGENT(H, NodeAgentList, LineAgentList)
ϵ ← 0.0001
while TRUE
    do
    WAITNEXTMARKETROUND()
    SEND(BIDREQ, NodeAgentList)
    D ← RECEIVE(BIDS, NodeAgentList)
    λ ← EQUILIB-PRICE(D)                > First guess.
    > Θ: row vector of θi values.
    Θ[i]                                ← 0, i = 1 . . . Nl  > First guess.
    repeat
    > λ: search for commodity balance.
    λold ← λ
    λ ← FINDZERO(F-LAMBDA(λ, Θ, H, D), λ)
    δλ ← ABS(λold − λ)
    > Request new θ from line agents.
    SEND(THETAREQ, LineAgentList, λ, Θ)
    Θold ← Θ
    Θ ← RECEIVE(THETAS, LineAgentList)
    δΘ ← MAX(ABS(Θold − Θ))
    until MAX(δλ, δΘ) < ϵ
    P = λ + Θ ∗H                        > array of nodal prices
    > Communicate prices to the node agents.
    for k ← 1 to Nn
    do
```

```
SEND(PRICE, NodeAgentList[k], P[k])
▷ Signal market round end to line agents.
SEND(MARKETREADY, LineAgentLst)
```

We start the description of the code by making some general remarks important for all given code:

- **Agent Communications:** The agents communicate using the message passing procedures SEND and RECEIVE. The first takes a message ID as a first parameter (e.g. BIDREQ), a (list of) Agent ID(s) as a second followed by an optional list of parameters to be send along with the message. RECEIVE blocks operation until the specified message is (or messages are) received. It has two possible forms, receiving either a message of one single agent or receiving messages from a list of agents. The latter form returns the received parameters in an array.

- **Demand Function:** The demand function data structure is not specified in detail. A possible form is an array of tuples (p, d). For computational reasons, the chosen structure must allow for fast aggregation of demand functions by adding price-wise. Evaluation of a demand function d for a given price is denoted in the pseudocode as $d(p)$. For the tuple-based data structure, this would involve interpolation between two tuple values. For reasons of simplicity this is not included in the pseudocode.

- **Root Finding:** The procedure FINDZERO implements a univariate root finding algorithm. The call:
  $x \leftarrow$ FINDZERO$(F(x, y, z), x_0)$
  searches for a root of the function F with x as free parameter. The search starts at x_0 and parameters y and z are considered to be constant during the search.

The AUCTIONEERAGENT requests for the demand function of all NODEAGENT instances at the beginning of each market round. Using these functions and the network transfer matrix H , which is given to as a parameter to the Auctioneer, a search for λ and θ_i , $\forall i$ is started. As a first guess λ is set to the general equilibrium price and all θ values are set to zero. In the **repeat** loop the agent consecutively searches for the λ value that gives commodity balance for current θ values and requests the Line Agents for θ updates. When this optimization ends, the nodal prices are sent to the individual Node Agents.
The objective function for the λ optimization is:

```
F-LAMBDA(λ, Θ, H, D)
▷ Calculate nodal price vector P
P = λ + Θ * H
▷ Swing-nodal demand from load flow.
d*1 ← Σ_{k=1}^{Nn−1} D[k](P[k])
▷ Swing-nodal demand from demand function.
d*2 ← D[Nn](λ)
▷ Goal: d*1 equal to d*2
return d*1 − d*2
```

The Node Agents are assumed to operate some process that produces or consumes the commodity. Their demand function will be influenced by the state of that process. Upon request by the Auctioneer, the agents compose their bid and send it to both the Auctioneer and all Line Agents.

The Line Agents use the bids for calculating their expected line flow.

```
NodeAgent(AuctioneerAgent, LineAgentList)
while True
do
Receive(BidReq)
d = ComposeBid(ProcessState)
Send(Bid, AuctioneerAgent, d)
Send(Bid, LineAgentList, d)
p ← Receive(Price)
ConsumeAllocation(d(p))
```

The pseudo code for the LineAgent is:

```
LineAgent(i, H, z_{i,MAX}, AuctioneerAgent, NodeAgentList)
> Select the 'own', i-th, row from H
H_i ← H[*, i]
while True
    do
    D ← Receive(Bids, NodeAgentList)
    repeat
    [λ, Θ] ← Receive(ThetaReq)
    > Calculate nodal price vector P
    P = λ + Θ * H
    > Calculate the line flow z
    z_i = H_i * D(P)
    if |z_i| < z_{i,MAX}
    then θ_i ← 0
    else > find θ to solve overload
    θ ← FindZero(F-Theta(θ_i, i, λ, Θ, H_i, D, z_{i,MAX}), Θ[i])
    Send(Theta, AuctioneerAgent, θ)
    until PeekNextMsg() = MarketReady
    > Consume peeked MarketReady message
    Dummy ← Receive(MarketReady)
```

The objective function for the θ optimization is given by:

```
F-Theta(θ_i, i, λ, Θ, H_i, D, z_{i,MAX})
    > Calculate line flow z_i for this θ_i
    Θ[i] ← θ_i
    P = λ + Θ * H
    z_i = H_i * D(P)
    > Goal: |z_i| equal to z_{i,MAX}
    if z_i > 0
   then return z_i − z_{i,MAX}
   else return z_i + z_{i,MAX}
```

A.2 Example

Figure A.1 gives an example algorithm outcome. Each of the lines has a capacity constraint of 1. All demand is located in the four nodes to the far left, while all supply is at the four nodes far right. All demand functions are S-shaped (i.e. sigmoidal) with the inflexion point at varying price levels between 5 and 14. The maximum demand per node is 2 for consuming nodes and −2 for producing nodes. It can be seen from the figure that in the network feasible (NF) solution the nodal prices are such that neither of the line flows exceeds the limit of 1. In the general equilibrium (GE) solution, six of the 14 lines are overloaded as shown in table A.1.

Table A.1: Line flows for the General Equilibrium and Network Feasible solutions.

	z_1	z_2	z_3	z_4	z_5	z_6	z_7
GE	1.4	0.4	0.6	0.8	0.4	1.8	0.2
NF	1.0	0.4	0.6	0.4	0.2	1.0	0.4
	z_8	z_9	z_{10}	z_{11}	z_{12}	z_{13}	z_{14}
GE	2.4	0.8	0.5	2.1	1.2	0.3	1.2
NF	1.0	1.0	0.7	1.0	1.0	1.0	1.0

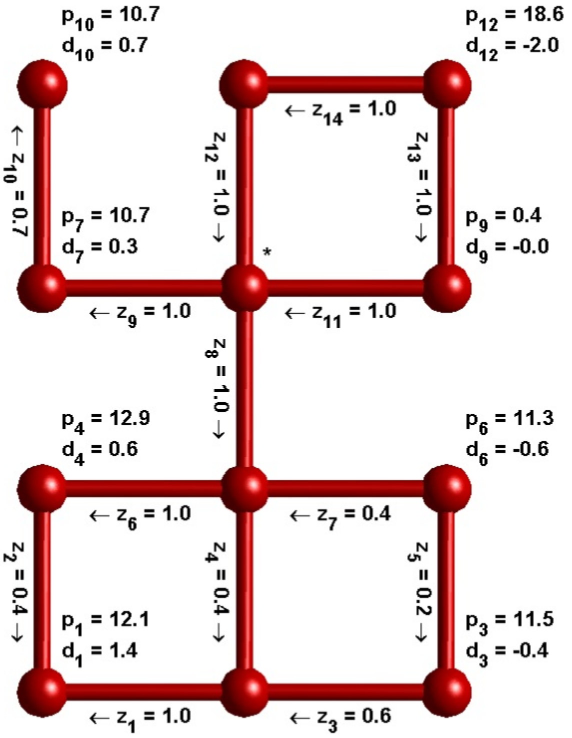

Figure A.1: Example network feasible market result. The maximum line capacity is set to 1 for each line.

A.3 Fast LMP in acyclic networks

Most distribution networks are operated radially. This means they are tree-shaped,. Their topology shows no cycles. In this section, we present a fast algorithm for calculating LMP solutions in such acyclic networks.

The algorithm is based on propagating demand functions from the leaves of the tree to the root (swing node) in the first phase and back-propagating the nodal price to the leaves in the network characteristics: line capacities and transport losses.

Four different algorithmic operators can be distinguished, one concentrating incoming demand functions at the network nodes, one for propagating demand functions along network lines towards the swing node, one for determining the nodal price at the swing node, and one propagating price information back to the leaves determining the nodal prices on the way.
The agents communicate using message passing procedures SEND and RECEIVE. The first takes a message ID as a first parameter (e.g. BIDREQ), an Agent ID (List) as a second followed by one or more optional parameters to be send along with the message. RECEIVE blocks operation until the specified message is (or messages are) received. It has two possible forms, receiving either a message of one single agent or receiving messages from a list of agents. The latter form returns the received parameters in an array.

LeafAgent

The pseudocode of the LEAFAGENT is quite straightforward. After reception of a bid request, it composes its bid according to its current preferences. The bid is sent off to the Line Agent associated with the line connected to the agent's node. After reception of the resource price the agent consumes its allocated power given by $d(p)$.

```
LEAFAGENT(Line)
while TRUE
    do
        ✄ First Phase:
        RECEIVE(BIDREQ, Line)
        d = COMPOSEBID(Preference)
        SEND(BID, Line, d)
        ✄ Second Phase:
        p ← RECEIVE(PRICE, Line)
        CONSUMEALLOCATION(d(p))
```

The Line Agent

The line agent implements the propagation. In the first phase the demand functions are transformed according to the local network characteristics: line capacities and transport losses.

```
LINEAGENT(HeadNode, TailNode)
while TRUE
    do
        ✄ First Phase:
        a ← RECEIVE(BID, TailNode)
        ā ← PROPAGATE(a)
        SEND(BID, HeadNode, a)
        ✄ Second Phase:
        p_j ← RECEIVE(PRICE, HeadNode)
        p_k ← PRICEBACKPROP(p_j)
        SEND(PRICE, TailNode, p_k)
```

The propagate operator propagates a concentrated demand function at a given node v_k over the first line, denoted e_i, in the path between v_k and the root, i.e. v_k is the tail of e_i. The operator accounts for the line capacity and network losses. The propagated demand function $a_k(p)$ for a line capacity constraint is calculated as:

$$\bar{a}_k(p) = \begin{cases} z_{i,\max} & z_{i,\max} < a_k(p) \\ a_k(p) & -z_{i,\max} \leq a_k(p) \leq z_{i,\max} \\ -z_{i,\max} & a_k(p) < -z_{i,\max} \end{cases} \tag{A.1}$$

while the propagated demand function for line losses can be calculated as

$$\overline{a}_k(p) = a_k(p) + l_i a_k^2(p) \tag{A.2}$$

where l_i is the loss factor, which is a function of the line resistance r_i. Naturally, the propagation in (A.1) and (A.2) can also be combined in one operator.

```
NODEAGENT(ChildNodeList, ParentNode)
while TRUE
    do
        ✄ First Phase:
        A ← RECEIVE(BID, ChildNodeList)
        a_k ← CONCENTRATE(A)
        SEND(BID, ParentNode, a_k)
        ✄ Second Phase:
        p_k ← RECEIVE(PRICE, ParentNode)
        SEND(PRICE, ChildNodeList, p_k)
```

Pseudocode for ROOTAGENT:

```
ROOTAGENT(ChildNodeList, LeafAgentList)
while TRUE
    do
        ✄ First Phase:
        WAITNEXTMARKETROUND()
        SEND(BIDREQ, LeafAgentList)
        A ← RECEIVE(BID, ChildNodeList)
        ✄ Second Phase:
        a_n ← CONCENTRATE(A)
        λ ← FINDROOT(a_n)
        SEND(PRICE, ChildNodeList, λ)
```

Concentrate

The concentration operator concentrates for a given node v_k the local demand function $d_k(p)$ and the demand functions $a_i(p)$ propagated into the node. The node receives incoming demand functions from all connected lines directed away from it. The concentrated bid at node v_k is calculated as:

$$a_k(p) = d_k(p) + \sum_{i:e_i \in Y_k} \overline{a}_i(p) \tag{A.3}$$

where $a_i(p)$ is the demand function propagated to v_k over line $e_i \in Y_k$. Y_k is defined as:

$$Y_k = \{e_i | h_i = v_k\} \tag{A.4}$$

the set of directly connected lines directing away from the root of G.

Swing Nodal Price

The price at the swing node (v_1) is chosen such that the market at the swing node is in equilibrium:

$$p_k = \begin{cases} p_j & a_k(p_j) = \overline{a}_k(p_j) \\ p_k : \overline{a}_k(p_k) = z_{i,\max} & a_k(p_j) > \overline{a}_k(p_j) \\ p_k : \overline{a}_k(p_k) = -z_{i,\max} & a_k(p_j) < \overline{a}_k(p_j) \end{cases} \tag{A.5}$$

where $a_1(p)$ is the concentrated demand function for the swing node.

Price Back Propagation

The price at the swing node is then propagated back along each line in the tree network using the price-back-propagation operator. This operator determines the nodal price for each node v_k. Consider a node v_k directly connected by line e_i to node v_j, with v_j the head and v_k the tail of the line. Then, v_k gets its nodal price back-propagated from node v_j. When v_j has a back-propagated price p_j, then for the propagation in (A.1), p_k is calculated as:

$$p_k = \begin{cases} p_j & a_k(p_j) = \overline{a}_k(p_j) \\ p_k : \overline{a}_k(p_k) = z_{i,\max} & a_k(p_j) > \overline{a}_k(p_j) \\ p_k : \overline{a}_k(p_k) = -z_{i,\max} & a_k(p_j) < \overline{a}_k(p_j) \end{cases} \tag{A.6}$$

and for the losses in (A.2):

$$p_k = p^* \left(1 + l_i a_k(p^*)\right) \tag{A.7}$$

Propagation of non-network constraints

Many other constraints can also be introduced. For example, grid operators want to minimize the aging behavior of a transformer, because it degrades faster for higher loads, which decreases its lifetime. The investment costs for such a replacement can be postponed if peak loading of the station can be avoided. A transformation on the demand function can be used to charge degradation costs to the end-customer. Hence, the end-customer gets a financial incentive to shift its demand from times of peak load to times of off-peak load. Another example, a household must pay taxes over the imported and exported electricity. If the supply and demand within the household is matched more frequently, the net import and export is reduced, decreasing the amount of taxes to be paid. This is especially financially beneficial if market clearing prices show relative small fluctuation compared to the tax rate.

Appendix B. PowerMatcher Simulation Tool Manual

PowerMatcher Simulation Tool

http://www.powermatcher.net/

Version	Author	Comments
0.1	G.M. Venekamp	Initial version
0.2	G.M. Venekamp	Revised version based on user input
0.3	G.M. Venekamp	Updated documentation to the latest version of the simulation tool
0.4	G.M. Venekamp	Small updates to reflect most of the latest changes

Contents

1 Introduction

The PowerMatcher Simulation Tool is made for simple demonstrations of the PowerMatcher technology made by ECN. Although it can show graphs, it is not meant as an analysing tool. For doing large scale analysing, the tool is able to retrieve data from the Agents that they make public and write these data to disk. The PowerMatcher and the PowerMatcher Simulation Tool are developed at ECN.

2 PowerMatcher Simulation Tool

2.1 prerequisites

There are a few prerequisites before you can successfully start the Simulation tool:

- Administrator rights. The user that will invoke the PowerMatcher Simulation Tool needs to have administrator rights as the PowerMatcher framework needs to start a few services;

- A fully initialized network card. This means an IP address needs to be assigned to it;

- Microsoft .NET Framework 3.5 SP1 needs to be installed;

2.2 Installation

Start the executable and follow the instructions given by the wizard.

3 Starting the simulation tool

The PowerMatcher simulation tool can be started by accessing the start menu. After starting the tool, you will be presented with a window as shown in Figure 1. This window is divided in three sections. On the far left, you will be able to create your scenario. It views you scenario in a list like manner. The “middle” part displays information about nodes in your cluster and their properties. Nodes can be agents, concentrator, auctioneers, global timers and price basis. Here you can adjust, or configure if you wish, the properties a node exposes, as well as selecting what information to view in a graph, i.e. what data an agent publishes to be viewable in a graph. These graphs are displayed on the far right of the window.

The first thing you will need to do is either create a scenario, or load an existing one. This scenario will be used by the tool to start the PowerMatcher cluster, i.e. auctioneer, network officers, concentrators and agents.

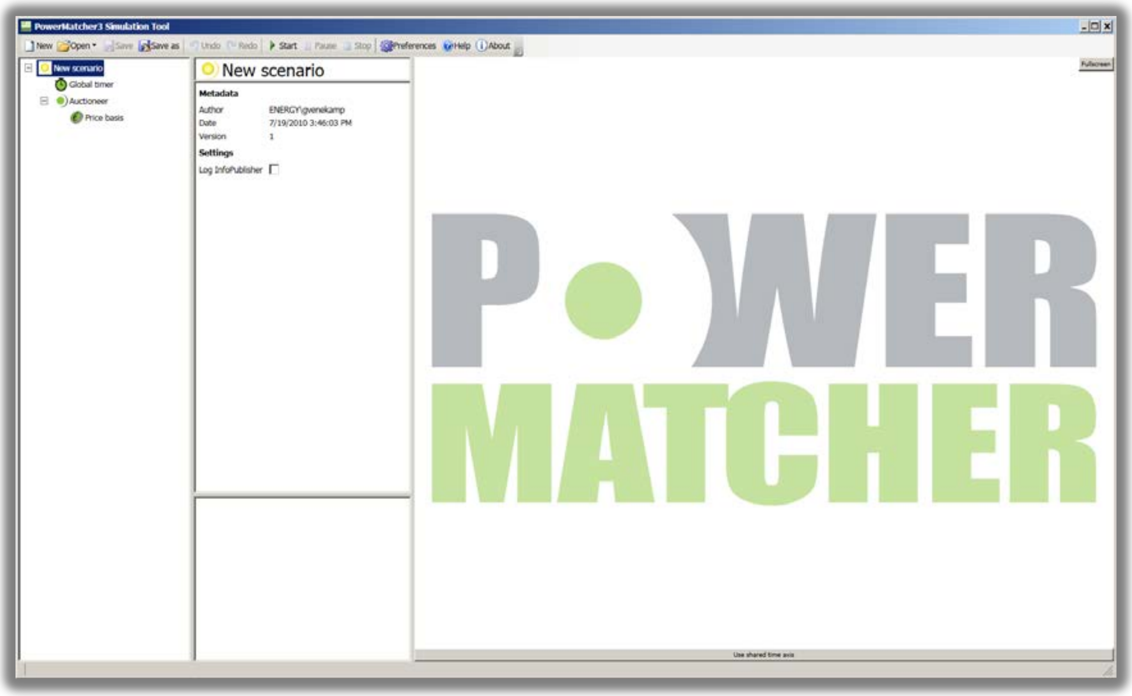

Figure 1: Opening screen of the PowerMatcher Simulation Tool.

3.1 Saving and opening scenario files

At any time you are able to quit the PowerMatcher Simulation Tool. When you quit the tool and you have made changes to the current scenario, you will be prompted to save your changes. You can also use one of the save buttons located at the top of the screen.

When you want to load an existing scenario, use the open button. The simulation tool uses a default location for scenario files. Refer to Appendix A on page 14 for the location.

3.2 Creating a new scenario

A new scenario is created by clicking the `new` icon on the top the screen. You will see an update of your window as shown in figure 2 on the next page. The name of the scenario is changeable and is called “New scenario” by default. To change the name, refer to Figure 3 on the following page. First click within the red circle and then edit the name within the green one after clicking there. Changing the scenario name to something logical is highly recommended. The simulation tool writes the data to files so you can use the data for further analyses. It does this in a directory with the same name as your scenario. The full name can be found in Appendix A.

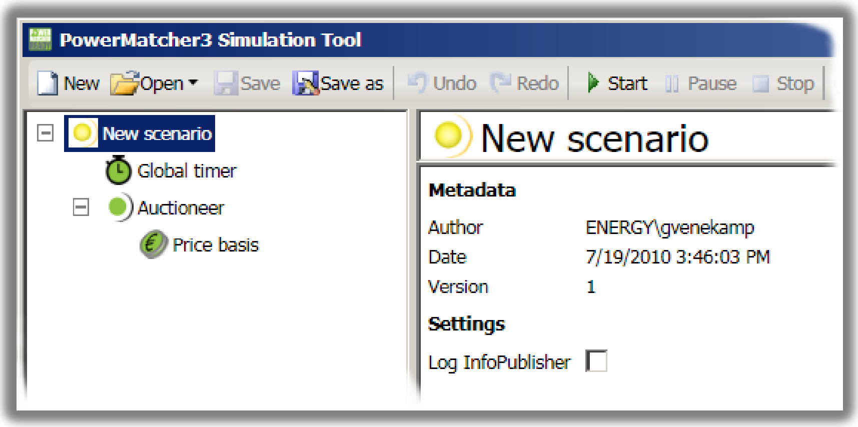

Figure 2: Part of the window after starting a new scenario.

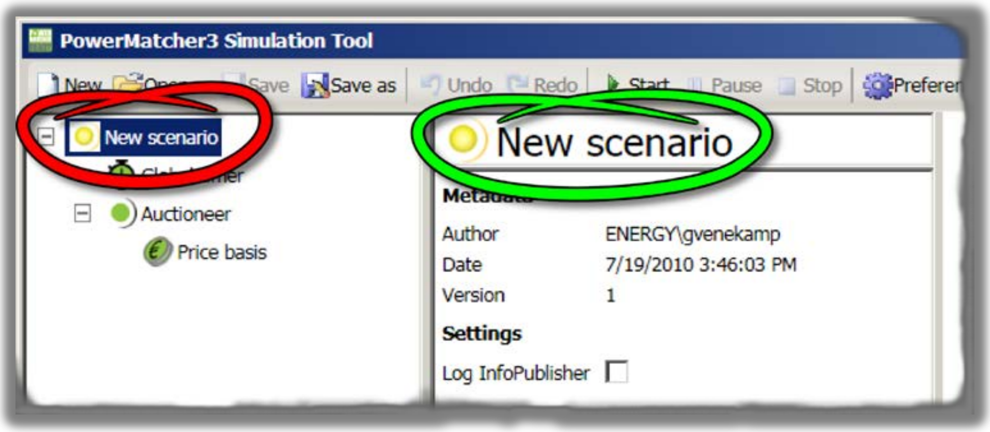

Figure 3: Changing the name of a scenario.

3.3 Global Timer

Clicking on the `Global Timer` on the left side of the window, will display the properties of the global timer. It has two intervals, a real and a virtual interval as well as a start date. The start date is only of importance to the agents. It is possible that agents use a data file in which real data is present, e.g. a date-time stamp is used. If the date in the data file start at 2010-01-01 and the simulation date is set to 2000-01-01, it will take quite some time before that particular date is reached. As long as the simulation date has not caught up with the date as found in the data file, the agent will happily wait for the simulation date to catch up and do nothing in the meantime. To avoid such problems, make sure the dates match.

The two other items you can change are the time intervals. The `Real Interval` specifies the time in HH:MM:SS format between updates in the network, i.e. the amount of time between two successive commands to simulate a time step. Typical values are in the range of 500–1000 milliseconds for non-complex Agents. If the `Real Interval` is 500 ms, then two simulations steps are taken for each passed second on *your* clock. However, do take into account the processor speed and the load on your CPU. This is what determines the minimal interval at which your simulation can run.

One can also use a zero time interval for the `Real Interval`. Doing so makes the simulation run as fast as possible. This is usually the preferred mode as it save you from tweaking the optimal value for the interval.

The `Virtual Interval` specifies in milliseconds the amount of time that progresses between successive real intervals, i.e. the date/time is advanced with this amount for each simulation step. Typical values could be one, five, ten or fifteen minutes.

Let's take 1 s as an example for the `Real Interval` and ten minutes for the `Virtual Interval`. This means that at each second of *your* time, the simulated time inside the simulator is advanced by ten minutes. So, it would take $\frac{24\times60}{10}$ = 144 seconds to simulate a whole day.

3.3.1 Simulation speed and the Global Timer

The settings of the global timer determine the speed at which your simulation runs. To speed up the simuation you have three options:

1. increase the `Virtual Interval`, thus doing more processing per second;
2. decrease the `Real Interval`, i.e. decreasing successive time steps the simulation takes (time seems to pass by a lot quicker to an Agent, where it does not when the `Virtual Interval` is increased);
3. The third and best option is to set the `Real Interval` to zero and have the simulation run at its maximum speed and choose the appropriate `Virtual Interval` according to your needed time resolution.

The `Global Timer` has also the ability to run in real-time. When set in real-time, the `Virtual interval` is ignored and the real date and time is used instead. See Section 3.4.1 for more information on the use of this property.

3.4 Auctioneer

The Auctioneer has a few properties as well. One of them is `Price update delay`. It specifies in milliseconds the amount of time the auctioneer has to wait for it to publish the new price. Agents will not send their bids at exactly the same time and in order to avoid many price updates, the auctioneer can be told to wait a bit. In the meantime the auctioneer will collect all bids and hopefully has collected all bids by the end of the waiting period. This should result in sending out one price signal.

One important note to make here is that the time specified here is always in real time! This means that if you specify a value of 100ms, the Auctioneer will wait 100 real milliseconds. If you have a fast running simulation it might by the case that the simulation time has progress quite a bit during those 100ms. From the perspective of your simulation it might look like that no price updates are send for hours and thus your simulation could be influenced by selecting large waiting periods. It is advised to use the default value of 1ms as this usually gives too little time for all necessary processing and thus is acts as though the prices are sent after a bid curve is received, i.e. after each received bid curve, a new price is sent immediately.

3.4.1 Connecting external Agents or Concentrators

It is possible to have an external Agent or Concentrator connect to a simulation. The Port property specifies a port at which external elements can connect to. The network service is created at the configured port, and the matcher itself at port+1. You can let the external agent connect to the network service’s two endpoints:

1. `http://localhost:PORT/IAgentManagementService`

2. `http://localhost:PORT/IDirectoryFacilitatorService`

However, when using an external element, the `Real-time` property must be ticked.

3.5 Price basis

It is best to leave the price basis at its default settings for now. The price basis specifies the initial minimum and maximum prices as well as the granularity of the interval in terms of price steps. The `Significance` specifies to which degree changes in price signals are considered significant. The number tell how many digits after the floating point are take into account.

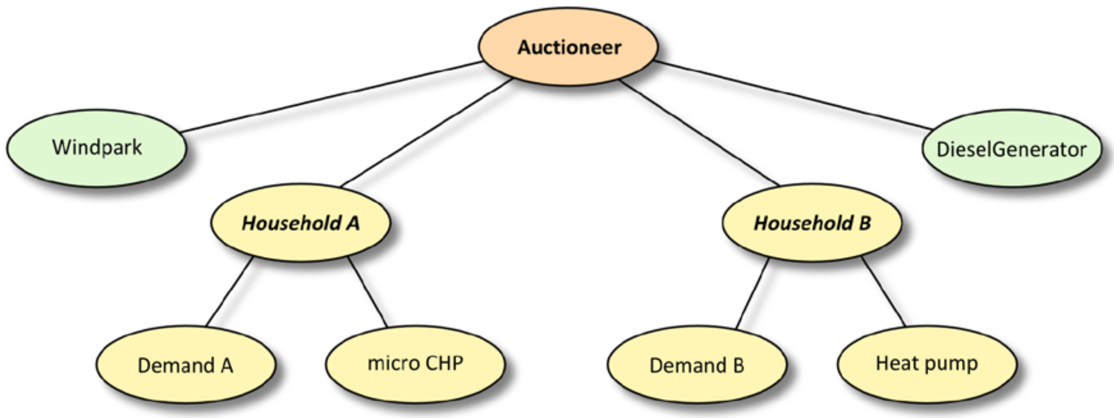

Figure 4: An example of a simple PowerMatcher cluster.

4 Creating a scenario

You can add any number of agents or concentrators to the auctioneer. Agents are end points and thus nothing can be added to them. Concentrators however, can have any number of agents and concentrators.

Let us assume that you want the following cluster. A wind park and diesel generator, two households, one with a heat pump, the other with a micro-CHP. Both households have a demand. Figure 4 shows the cluster in a graphical manner.

4.1 Adding top level agents

We will start by adding agents for the wind park and diesel generator. They should be added directly to the auctioneer. Right click on the auctioneer label and then a context menu appears which lets you add an agent or concentrator. Refer to Figure 5 for the context menu. Adding elements to concentrators works exactly the same as adding elements to an auctioneer.

What you could change is the default name of the agent in the name field. In our case the `WindTurbineAgent` and `Dieselgenerator`. Add them both to the auctioneer. The names used are 'Wind Park' for the wind turbine and 'Generator' for the diesel generator. When done you should have something like figure 6 on the following page.

4.2 Adding the clusters

What needs to be done next is adding two clusters. As can be seen in Figure 4 we need to have a concentrator for each household. Add two concentrators at the auctioneer level, just like you have added the agents. If you wish you could rename the concentrators to a more logical name like 'Household A' and 'Household B'. This is done by editing the text box as shown in Figure 7 on page 8. You have to select the concentrator of which you want the name to be changed first on the left part of the window. Also change the `PriceUpdateDelay` for both concentrators to a value of '1'. Do not use '0', as this value is illegal.

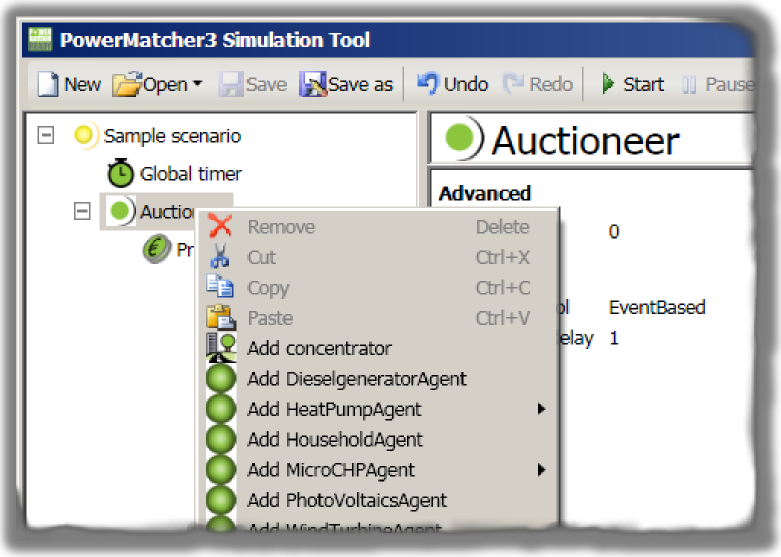

Figure 5: Adding agents and concentrators.

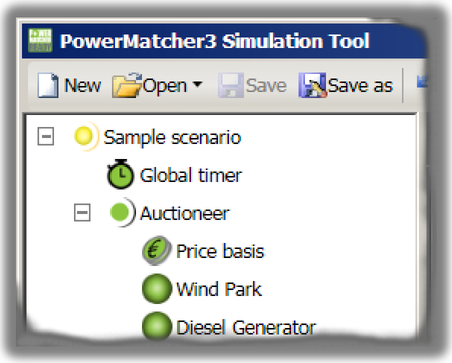

Figure 6: View of the PowerMatcher Simulation Tool after adding a Wind Park and a diesel generator.

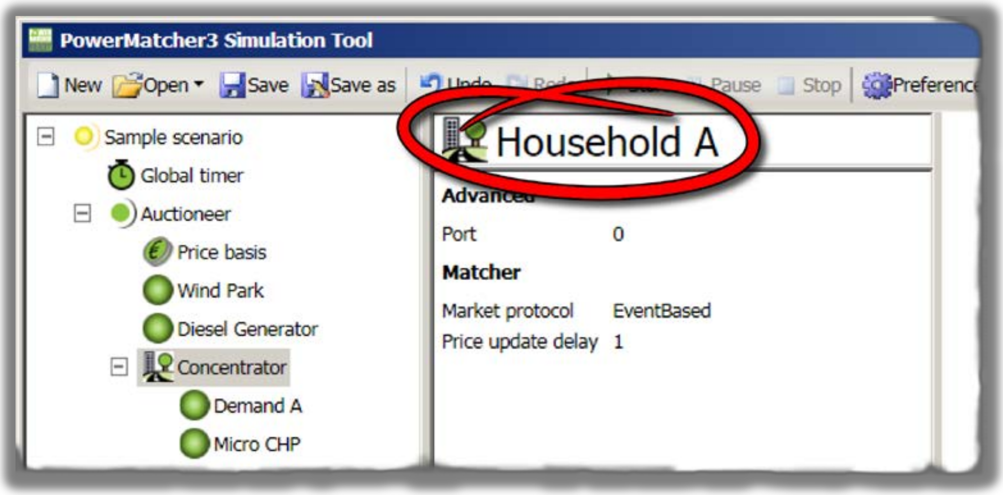

Figure 7: Changing a name of a concentrator in the tool.

We are not finished yet, the newly created concentrators need to a have two agents below them. Adding agents to a concentrator is the same as adding agents to the auctioneer. Right click on the concentrator you want to add an element to.

Some agents support different controllers and you can't add such agents without selecting a controller. You will see an expansion marker to indicate that you have a choice in controllers. See Figure 8 on the next page for an example. `Default` controllers are the ones that will try to make intelligent bids.

For each concentrator add an `HouseholdAgent` and give it a local name like Demand A, or Demand B. To have a different demand profile for each household, change the name of the Data file property for 'Household A' to `.\HouseholdSampleData-1.txt` and to `.\HouseholdSampleData-2.txt` for 'Household B'. Now you need to add a `MicroCHPAgent` for 'Household A' and a `HeatPumpAgent` for 'Household B'.

As a final step to finish creating this example scenario, you should change the `Random seed` in both the heat pump and micro-CHP agents. You can find this setting under the 'SpaceHeatingBuffer' of either agent. This ensures that the building model parameters differ, e.g. initial room temperature. Make sure the two seeds are not the same.

Now that you have created your first scenario, it is time that it is saved to disk. For this, use the `Save as` button on top of the window. The Default location presented to you should be fine.

4.3 Removing a node

If you wish to remove either an agent or a concentrator, you can do so by using the right mouse button to bring up the context menu. Be sure to select the agent or concentrator you want to be removed first. Or you can select the element you want to remove and use the `delete` button to remove it.

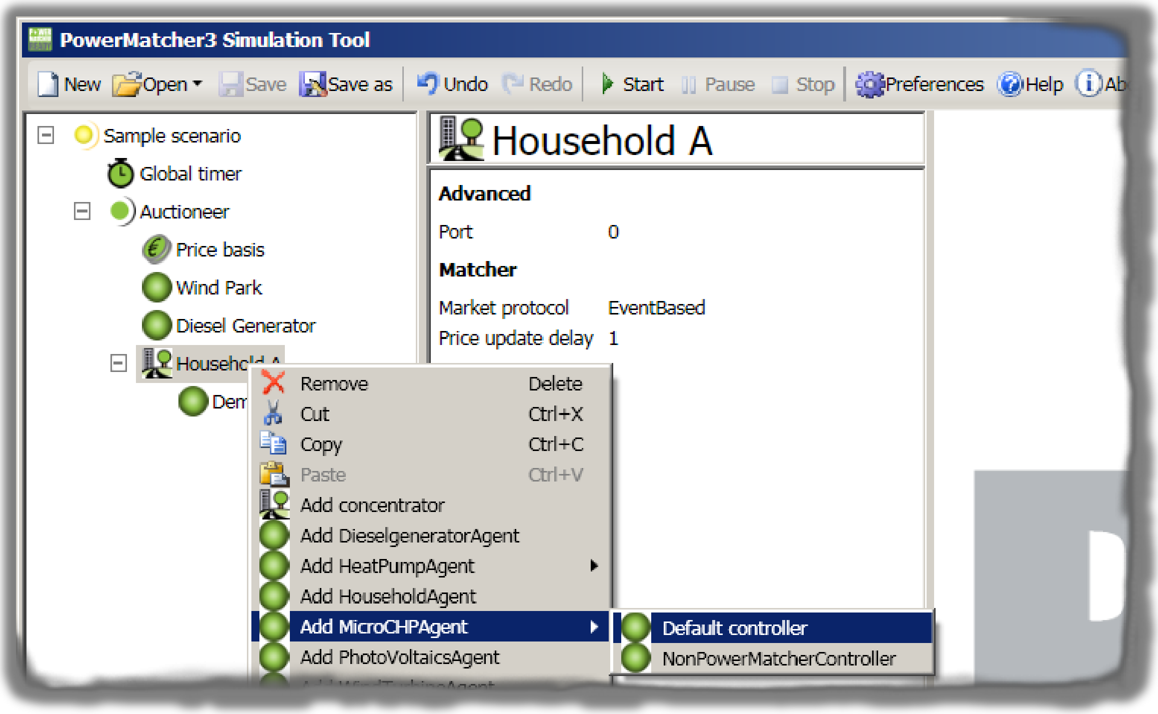

Figure 8: Adding a micro CHP agent with the default controller.

4.4 Configuring the agents

The agents you have added to your scenario might need some additional configuration. For example you might want to give certain agents a data file. In our case the two household could have different data files for no household will have exactly the same demand profile. Also, the wind park might be too large or too small to our taste and this too can be adjusted. This is done in the middle part of the window, i.e. right beside our scenario layout and where we already have changed the names of the household concentrators. However, explaining what the agents can do and what properties can be configures is outside the scope of this document and one should refer to the documentation of the individual agents.

5 Starting a simulation

It is time to start a simulation with our scenario. This is accomplished by pressing the start button, Figure 9 on the following page shows the location of the `start` button. It is on the top of the window. After having started the simulation, the `start` button becomes inactive and the `pause` and `stop` buttons located to the right of the `start` button become active. They can be used to pause and stop the simulation.

In the scenario description you will see the label changing colours. When agents are being started, they become bluish first and only when they have been successfully started, green.

Figure 9: Location of the start button.

Should they turn red instead, something has gone wrong and the agent was not started. The middle section on the bottom of the simulation tool windows a small hint is presented in such cases about what might have gone wrong. You should fix the errors first before you can successfully start the simulation.

6 Displaying graphs

The PowerMatcher Simulation Tool does not know what data is being published by the agents when it is started. It will have to wait until the simulation is started and the agents have made known what they publish. Only then can the simulation tool start to display the published data. In order to know what data is being published, click on the agent you are interested in in the cluster overview on the left side of your window. The simulation tool shows the available information from an agent in the middle part of the window, towards the bottom. You can tick the data you wish to be displayed[1] and the tool will show the graph. It will only show data that is published *after* the moment you selected it. This means that it could take a little while before data is being shown in a graph. It depends on the frequency at which the Agent publishes its data. Also, all previous data that has been published will *not* be shown in the graph.

When you restart the same the simulation, the tool remembers what data is published and it also remembers the graphs it plotted last time. Thus, the plotting start immediately upon starting the simulation.

6.1 Multiple graphs

The simulation tool is able to show more than one graph as the time. Just tick the data from an agent or auctioneer you wish to view. Figure 10 on the next page gives you an example of how the graphs might look like.

6.2 Full screen graphs and graph synchronisation

On the graph area you will find two buttons. On is located at the top to right, see the red circle in figure 10 on the following page. Clicking it, makes the graph appear full screen. The other button is at the bottom of the graphs and can be used to force a synchronisation of the time axis across all visible graphs.

[1] The PowerMatcher Simulation Tool will only be aware of what an Agent publishes after the Agent has published its first data. This means that at the very start of the simulation tool, no data has been published yet and therefore you can't select any graph to plot. Wait until the Agent has published its first data to select the graphs you are interested in.

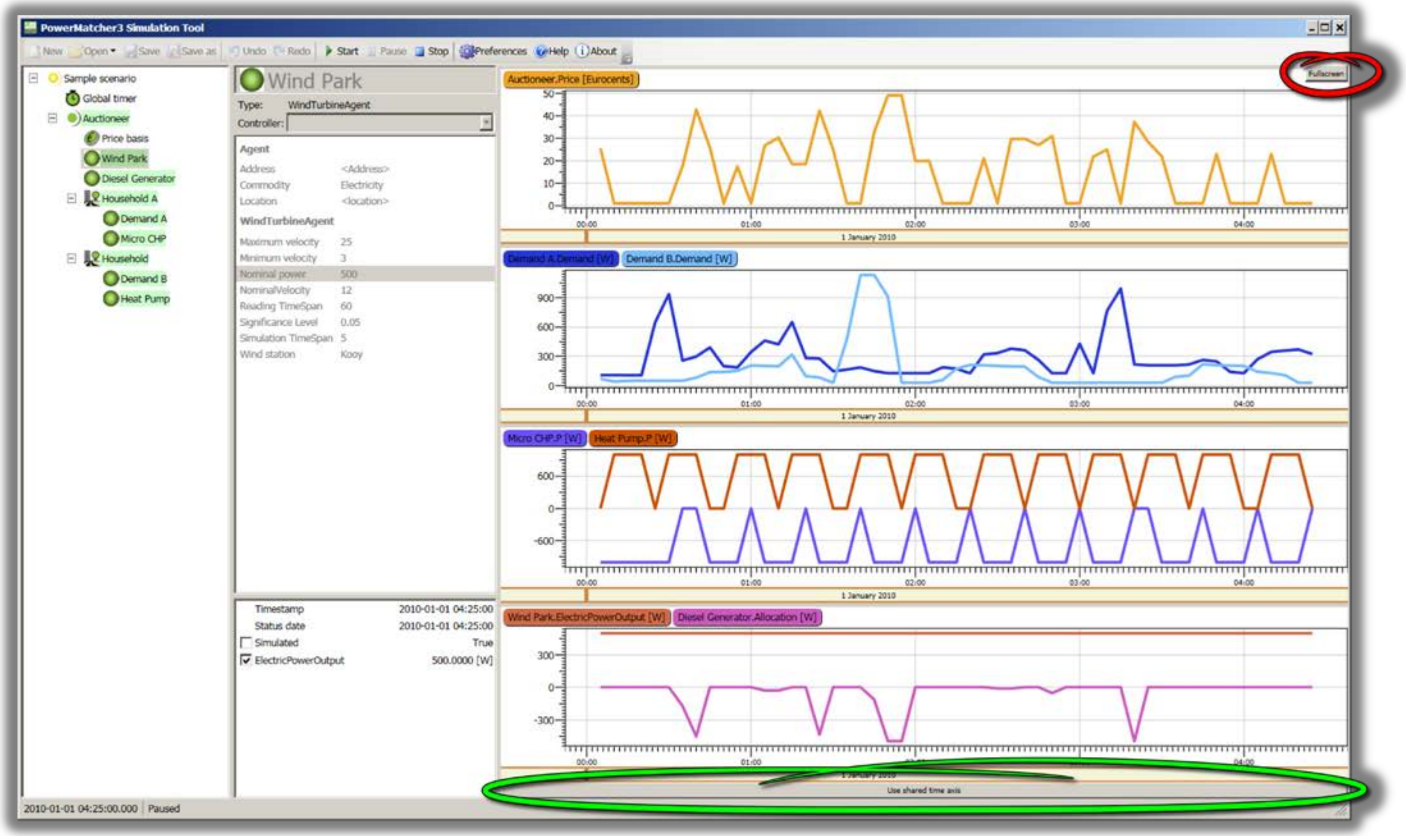

Figure 10: Example how graphs might look like in the PowerMatcher Simulation Tool. Here we see a price, the demand of both households in one graph, as well as the production of the micro CHP and consumption of the heat pump, the output of the wind park and the production of the generator.

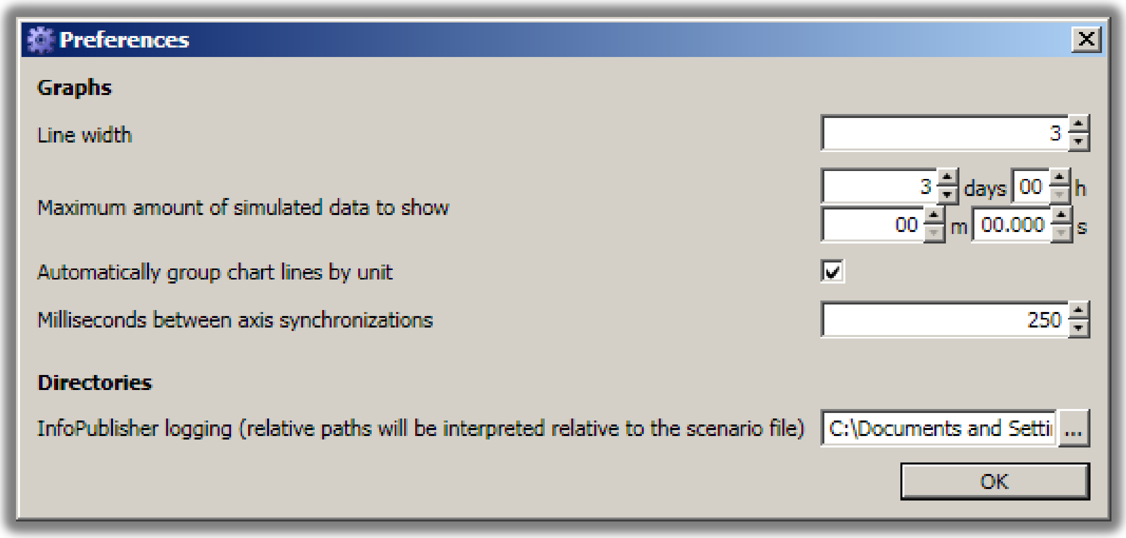

Figure 11: Preference settings.

6.3 Combining and removing graphs

Graphs can be combined in one plot. To accomplish this, drag one graph into another by grabbing the label of the graph you wish to move and drop it on the graph area of the graph you wish it to move to.

In order to remove a graph you have to grab the label of the graph you wish to remove. When grabbing, a drop area pops up in the lower right corner of the screen. Drag the label to there and drop it to have it removed.

7 Preference settings

The PowerMatcher Simulation Tool lets you select a few preferences. This can be accessed by pressing the `preferences` button on the tool bar. Figure 11 show the windows

that should pop up. The location where the InfoPublished stores its data is configurable and here you set the root directory.

8 Logging

Although normally not necessary, the PowerMatcher Simulation Tool logs what it is doing in one single log file. The framework that is used for logging is `log4net`[2] . The configuration file for `log4net` is found in the same directory as the tool and is called: `logconfig.xml`, see also Appendix A. By default the logging level is set to WARN, setting it to DEBUG will make it show more information. Note that your simulation will likely require more time to execute.

[2] `http://logging.apache.org/log4net/index.html`

Appendices

A File locations

Data files necessary for agents are resolved against the location where the scenario file is located and not from configurable InfoPublisher path.

By default the following directory location is used:

```
C:\Documents and Settings
  <your account>
    My Documents
      PowerMatcher3 Simulation Tool
```

It can be changed through the preference settings. Under this directory the following structure exists:

```
Data
  InfoPublisher
    <scenario 1>
    <scenario 2>
    <scenario 3>
...
Logs
```

Each time you run a scenario its data will be stored in a new directory under your scenario map. The name of this new directory will have a `date.time` format. Thus:

```
InfoPublisher
  Sample scenario
    20100721.094332
      Agent1.csv
      Agent2.csv
      Agent3.csv

    20100721.102705
    20100721.130524
```