Communication Requirements of Distributed Energy Management Algorithms in Smart Grids

Daniel Hölker · Daniel Brettschneider
Michael Sonnenschein · Ralf Tönjes

Abstract This publication compares three different distributed energy management algorithms. All algorithms are evaluated regarding quality of energy management and communication requirements. In addition, their scalability and behaviour at communication limitations are analysed. Furthermore, recommendations for the use of the different algorithms are given. The first algorithm is COHDA. It has a fully distributed approach without any central unit. Secondly, the well known algorithm PowerMatcher, which performs market based supply demand matching, is analysed. Thirdly, a round-based and privacy preserving algorithm called PrivADE is evaluated. All algorithms are simulated in the ns3-based simulation environment SiENA.

Keywords smart grid · energy management · communication requirements

1 Introduction

Energy management in the domestic area will become a vital part in the future power grid. This comprises Demand Side Management (DSM) and the management of supply units like micro Combined Heat and Power Plants (μCHPs). To handle the possibly high number of households and devices, different Energy Management Algorithms (EMAs) were developed. Their functionalities vary considerably and they partially pursue different goals. Specifically EMAs were developed for day ahead scheduling of μCHP or Heat Pump (HP). Other EMAs were made for frequency response by using Battery Storages (BSs) or Electric Vehicles (EVs). A third EMA application is intra-day load balancing with lower real time requirements than frequency response EMAs.

The convergence of the aforementioned day ahead scheduling algorithms is not time-critical because they can be executed beforehand. This leads to low restriction regarding convergence times and therefore low communication requirements. In contrast, frequency control algorithms have to react within very low delays (often less than 1 second) and very high reliability. All information that is needed is the grid-frequency which is inherently available through the power grid. An additional communication network would be redundant. The only kind of algorithms which should be analysed in perspective of communication requirements is the third group of EMA applications. Because, in contrast to day-ahead and frequency response EMAs, the behaviour of intra-day EMAs often depends on the communication network. So, the publication is focusing on requirements of this intra-day EMAs.

For these intra-day EMAs, various possibilities to manage households and their energy devices exist. A simple way is a central control unit, which controls each device directly. This method is called Direct Load Control (DLC). A more common and in the public more accepted way is an indirect management, e.g. by price incentives. For this indirect method, different EMAs were published in recent years. In this paper, three different algorithms are simulated and evaluated regarding communication requirements. The first algorithm is COHDA [3]. Originally, it was developed for day ahead scheduling of controllable power supply. How-

D. Hölker
Albrechtst. 30, 49076 Osnabrück, Germany
E-mail: d.hoelker@hs-osnabrueck.de

D. Brettschneider
E-mail: d.brettschneider@hs-osnabrueck.de

M. Sonnenschein
E-mail: michael.sonnenschein@informatik.uni-oldenburg.de

R. Tönjes
E-mail: r.toenjes@hs-osnabrueck.de

ever, COHDA was adapted in this paper to handle intra-day energy management of different devices. The second algorithm is PowerMatcher. It was first published by Kok in 2005 [5] and is mainly used for DSM in households. However, the concept of PowerMatcher can be also used for energy generating units. Additionally, we present PrivADE [1]. PrivADE is a round-based approach with homomorphic encryption to preserve users privacy.

There are already several evaluations that analyse communication requirements for smart grid applications. Saad [6] focuses on scheduling algorithms using game theory. He suggests Power-line Communication (PLC) or wireless technologies, but does not compare different algorithms. He highlights that the area of communications in smart grid systems is still in its infancy. Conejo [2] describes the importance to use a bidirectional communication, but does not analyse the requirements in detail. Samadi [7] also proposes a two-way communication. He compares the required amount of messages by his game theoretic approach to a price anticipating system. However, he does not compare their abilities with regard to energy management functions. So it remains unclear, if his game theoretic approach is advisable in all scenarios. Yan [13] describes challenges and requirements on communication in a smart grid. He gives a good motivation for communication in smart grids. However, he only provides an overview about required latency without focusing a concrete scenario.

Another overview is given by the US department of energy [10]. They categorise smart grid functionalities and give an overview of communication requirements. For demand response they estimate the required bandwidth between $14\,\mathrm{kbit\,s^{-1}}$ and $100\,\mathrm{kbit\,s^{-1}}$ as well as the latencies ranging from 500 ms up to several minutes. However, the functionalities that could be enabled with these communication properties are not described.

To enable a better overview, this paper focuses on required data amount and time for convergence of EMAs. This is simulated with households and their devices as controllable units in concrete scenarios. Especially scaling properties and behaviour with bandwidth limitations and high communication delay is analysed.

This publication is structured as follows. In section 2 the algorithms COHDA, PowerMatcher, and PrivADE are described. Afterwards, the simulation environment and the scenario is shown in section 3. In section 4, simulation results of the algorithms are shown and the communication requirements analysed. Finally, the algorithms are compared and recommendations for different scenarios will be given.

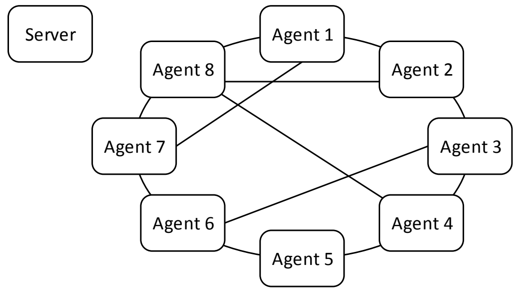


Fig. 1 COHDA - small-world overlay network example ($\phi = 0.5$)

2 Distributed Energy Management Algorithms

The communication requirements depend largely on the EMA itself. Several algorithms with different functionalities were published in recent years. In this section, the algorithms COHDA, PowerMatcher and PrivADE are introduced shortly to illustrate their functionalities.

2.1 COHDA

COHDA is a heuristic for multi agent systems [3]. Including our adaptation for the motivating use case of intra-day energy management in the present contribution, the algorithmic approach in COHDA ca be described as follows: Each agent represents a household $h \in H$. All households are connected by an overlay network and have identifications that are well-ordered. For best performance, the overlay network should be realized as a Watts-Strogatz small-world model [12] (see Fig. 1). Each household h has a predicted energy consumption C_h for the next interval and a total flexibility due to its adaptable devices. Furthermore, each household has a working memory and a solution candidate. Both contains its own planed energy consumptions and that of other households. Furthermore, the solution candidate stores the identification of its creator. The consumptions in the solution candidate corresponds to the actual consumption, if no better solution can be found. Whereas the values stored in the working memory, are used to search a better solution. The global goal of COHDA is to achieve the total goal consumption ζ and minimize the error-function $e(\zeta, \sum_{h\in H} C_h)$, which rate the deviation between goal and solution. COHDA works as follows:

1. The server initiates COHDA by sending a packet with the desired value $\zeta/|H|$ to a random household.
2. The household i that receives the first packet, chooses its own energy consumption C_i, which minimises the error-function. This value will be stored in the working memory κ_i and in its solution candidate

γ_i. Afterwards, the goal value, solution candidate and the working memory is sent to all neighbours in the overlay network.

3. A household i that receives a packet with a working memory κ_r and a solution candidate γ_r, firstly updates the energy consumptions in its own working memory ($\kappa_i \rightarrow \kappa_i'$). If it has been updated ($\kappa_i \neq \kappa_i'$):
   - If the amount of households in κ_i' is higher than the amount in γ_i and γ_r ($|H_{\kappa_i'}| > |H_{\gamma_i}| \wedge |H_{\kappa_i'}| > |H_{\gamma_r}|$), the best own consumption C_i^* will be selected (minimum $e(\zeta, C_i^* + \sum_{C_h \in \kappa_i' \setminus C_i} C_h)$), and κ_i' is set as a new solution candidate γ_i.
   - If the set of households in the received solution candidate is equal to the set in the own solution candidate ($H_{\gamma_r} = H_{\gamma_i}$):
     (a) If the received γ_r is better than the own γ_i ($e(\zeta, \sum_{C_h \in \gamma_r} C_h) < e(\zeta, \sum_{C_h \in \gamma_i} C_h)$), or γ_r is equal to the own γ_i ($e(\zeta, \sum_{C_h \in \gamma_r} C_h) = e(\zeta, \sum_{C_h \in \gamma_i} C_h)$) but has a solution creator with a higher identification, replace γ_i by γ_r.
     (b) Find the own consumption C_i^* that minimises $e(\zeta, C_i^* + \sum_{C_h \in \kappa_i \setminus C_i} C_h)$ and store C_i^* into κ_i. If κ_i has a lower error value than γ_i ($e(\zeta, \sum_{C_h \in \kappa_i} C_h) < e(\zeta, \sum_{C_h \in \gamma_i} C_h)$) replace γ_i by κ_i.
4. When either γ_i or κ_i have been modified in one of the previous steps, the household sends a new packet with the goal value, solution candidate and working memory to all neighbours in the overlay network.

When COHDA is converged, the predicted energy consumption can be set. Further information about COHDA is available in the Hinrichs publication [3].

2.2 PowerMatcher

PowerMatcher is a common method for supply demand matching. It was first published by Kok in 2005 [5]. In PowerMatcher, households send a bid to an auctioneer, which has information about the goal consumption and chooses a price depending on the accumulated bids. In the next sections, the methodology of creating bids and the execution of PowerMatcher is described in more detail.

2.2.1 Bids

Each adaptable device is represented by a device-agent. Every device-agent has to create a bid-curve, which depends on the environmental conditions. Example: a device agent for a HP adapts its bid-curve depending on the load level of the hot water tank (see Fig. 2). When the load level is high, the HP has not to run necessarily.

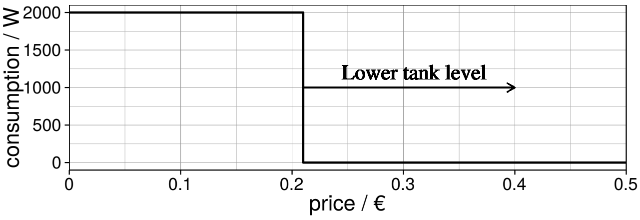


Fig. 2 Bid-curve of a heat pump

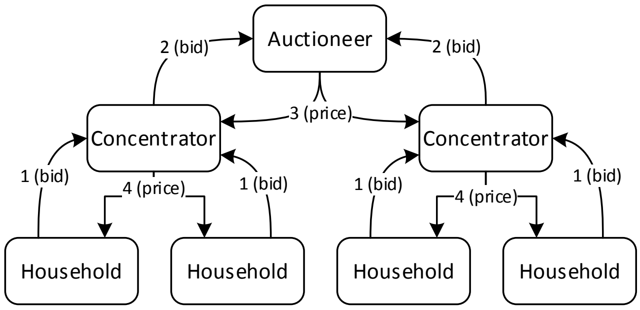


Fig. 3 Overlay network and steps during execution of PowerMatcher

In case of bid-curves for BSs or other continuously manageable devices, the bid-curve has no jump discontinuities. However, bid-curves are always monotonically decreasing functions. If the price increases, the consumption stays constant or will be lowered.

The bid of Fig. 2 can be represented by the coordinates {(0.21 €; 2 kW), (0.21 €; 0 kW)}. Values between the coordinates, are calculated by linear interpolation. Thus, continuous decreasing bids can be realised with only two coordinates too. The most bids of flexible devices have just two or four coordinates. That leads to a very small amount of data.

2.2.2 Execution

The execution of PowerMatcher consists basically of four steps, these steps are shown in Fig. 3 and described in the following:

1. Each device agent creates a bid. Each household aggregate these bids and send them to a concentrator.
2. All concentrators receive bids from different households. They aggregate these bids and send the result to another participant called the auctioneer.
3. The auctioneer calculates the price, where the bid-curve matches the total goal consumption ζ. This price is then sent back to the concentrators.
4. All concentrators receive the price and forward this to each household. The households set their devices to the corresponding consumption value.

2.3 PrivADE

PrivADE is a Privacy-Preserving Algorithm for Distributed Energy Management [1]. It is round-based and distinguishes adaptable loads that can be managed fine granular (BS and EV), and switchable loads which can only be turned on or off (μCHP, HP and heating rod). The households and the server are part of an overlay communication network that is arranged as a ring (see Fig. 4). The server knows the goal consumption ζ and tries to match the total consumption to this goal.

In the first round, all necessary data is gathered. Therefore, the server creates a data packet with several counters and sends it through the ring. Each household that receives this package adds its values to the corresponding counters. For example adds its total energy consumption to the corresponding counter. This is done using homomorphic encryption.

After the first round, the server has information about the total consumption C (e.g. 28 000 W), the amount of switchable devices with certain consumptions (e.g. two devices with 1000 W and one with 10 000 W can be turned on) and the possibilities of adaptable devices to increase (e.g. $\alpha = 4000\,\mathrm{W}$ by $A = 5$ households) or decrease the consumption (e.g. $\beta = -3000\,\mathrm{W}$ by $B = 2$ households). So the server decide, which device-categories (e.g. all devices with 1000 W) to switch, for allow achieving the goal consumption (e.g. $\zeta = 32\,500\,\mathrm{W}$) with the adaptable devices. So the server sends another packet through the ring with the devices to switch and a consumption share for the adaptable households (e.g. $\frac{32\,500\,\mathrm{W} - 28\,000\,\mathrm{W} - 2 \cdot 1000\,\mathrm{W}}{5} = 500\,\mathrm{W}$). Each household that receives this package adds, if appropriate, its adaptation and send the package to the next. If all adaptable households can fulfil their adaptation share, PrivADE has been converged.

If a household cannot fulfil its share (e.g. one household can only adapt to 300 W) the remaining households have to adjust their adaptations (e.g. additional $\frac{2500\,\mathrm{W} - 4 \cdot 500\,\mathrm{W} - 300\,\mathrm{W}}{5-1} = 50\,\mathrm{W}$). This requires another round. So the number of rounds can increase till maximum $|H|$ in the worst case.

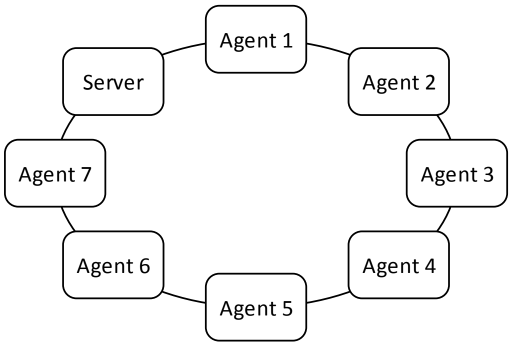


Fig. 4 Ring overlay communication network of PrivADE

3 Simulation Environment

To simulate the behaviour of the introduced algorithms, a lot of probabilities and surrounding conditions have to be considered. In the following subsections, the capabilities of our simulation environment called SiENA [9] and the scenario for our experiments are described. SiENA is integrated in the network simulator ns-3. This enables to simulate simultaneously the communication behaviour and the energy consumption.

3.1 Energy Simulation

SiENA contains a large data basis of energy consumption curves for the most relevant household appliances (stoves, office devices, washing machines, fridges, etc.). Market penetrations are specified by values of the German federal statistic office [8]. For realistic simulations, the simulator chooses appropriate activation times for the different devices. A simulated consumption and the German standard load profile (H0) match fairly well. This is shown in Fig. 5. Therefore, it can be assumed that the simulated energy consumptions are well grounded.

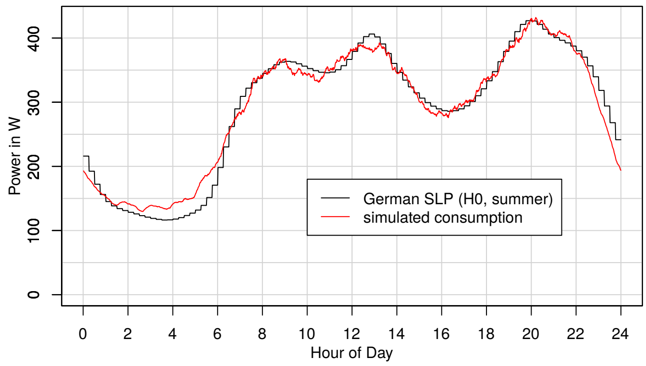


Fig. 5 Comparison of simulation and Standard Load Profile (SLP) for households

In addition to the devices commonly used today, devices like μCHPs and HPs can be simulated. Therefore, the heat demand according to the standard VDI4655 has been implemented [11]. In addition, EVs and BSs can be simulated. All these future devices have high potential for load shifting and load adaptation. The selection of controllable devices depends on the scenario, which is described in the next section.

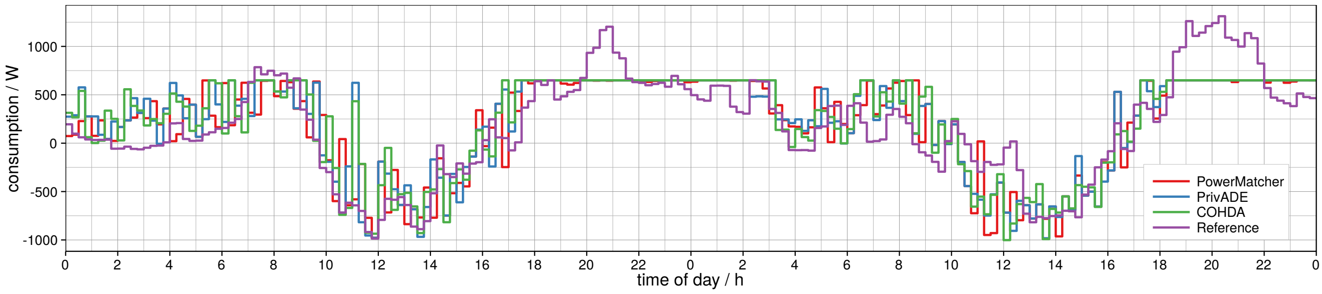


Fig. 6 Energy consumption of 50 households controlled by different algorithms

3.2 Scenario

Many scenarios exist that allow a useful energy management, e.g. load shaping and peak clipping (see Fig. 7). Load shaping can be used to adjust the energy consumption to fluctuations in generation. A fluctuating generation can be caused for example by renewable energy sources like photovoltaic systems or wind turbines. In our scenario, we assume a distribution grid supplies 50 households and a substation not allowing a higher total power consumption than 32.5 kW. So, algorithms for peak clipping are needed. However, simulations have shown that increasing the energy consumption before the peak can allow a better ability to reduce consumptions during the peak. So, the goal of the algorithms is to increase or decrease the peak demand to 650 W per household. This goal is typical for load shaping.

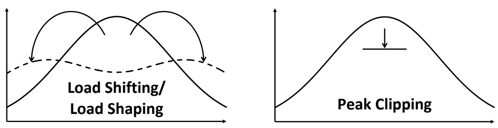


Fig. 7 Goals of energy management

In the scenario each household has probabilities for owning controllable devices. The probability to own a μCHP or a HP is 25 % each. Both devices are controllable if the corresponding heat storage is filled above 30 %. 20 % of households own an EV which is manageable if its load level is at least 90 %. Further, 30 % of the households own a BS. They have no special conditions for being controllable. Fig. 6 shows two exemplary days managed by each introduced energy management algorithm. It can be seen that all algorithms have similar capabilities to clip the peaks. The quality ratings according to the methodology of [4] are shown in Table 1 and confirm this observation.

Table 1 Quality ranking of different algorithms in percent (higher values correspond to a better result)

	COHDA	PowerMatcher	PrivADE
Peak Clipping	28	27	28
Load Shaping	25	23	25

3.3 Communication Network

Independent of the EMA overlay network, the communication technology has a physical topology. Therefore, a tree was selected, which can be found in wired internet connections like Digital Subscriber Line (DSL). Each household is represented by a leaf and upper level nodes represent network elements such as switches or routers. The maximum number of leafs for a node is ten and the graph has a maximum height of five, meaning the worst case path size from leaf to leaf is ten in case of more than 80 leafs. In case of more than 160 leaf nodes, the root has more than two connections (see Fig. 8).

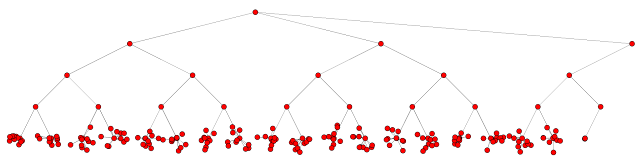


Fig. 8 Topology of a tree network with 182 nodes

The topology ends with households, meaning that no in-house communication is simulated. All leaf nodes (households and servers) have a 20 ms latency and a bandwidth of 5 Mbit s^{-1}. All other nodes (routers) have 2 ms processing delay and a data rate of 1 Gbit s^{-1}.

4 Communication Analysis

In this section, simulation results are described and discussed. The scenario of the previous section is used. Table 2 shows an overview of the simulation parameters. The algorithms use the UDP-Protocol for communication. All data amounts include the 30 bit MAC header.

Table 2 Overview of simulation parameters for the subsections 4.2 and 4.3

simulated objects:	50 households and their appliances
flexible devices:	μCHPs, HPs, BSs and EVs
simulation period:	varies depending on simulation complexity from 7 days up to 1 month
interval:	15 minutes
communication parameters:	leaf bandwidth $5\,\mathrm{Mbit\,s^{-1}}$, router bandwidth $1\,\mathrm{Gbit\,s^{-1}}$, delay 20 ms

4.1 Convergence Time

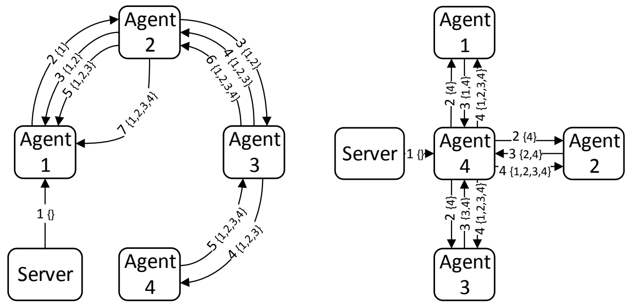
In this section, the required time for convergence is simulated and analysed. This is very important because the convergence time determines the interval within an algorithm can be executed. The smaller the interval, the faster the reaction of the energy management. Simulation period is one month. The figures in this section illustrate a curve for an exemplary day as well as a box plot for the entire simulation.

4.1.1 COHDA

COHDA is a heuristic and its convergence behaviour depends on various conditions. The best case for fast convergence occurs when no household has any adaptable device. So the households only inform the other households about their energy consumption. Another important condition is the overlay network. Because no representative small-world overlay network can be defined, the best case for an open ring and a star overlay network is described, as clarifying example in the following, before deriving properties of the small-world topology afterwards on this basis.

Fig. 9(a) shows the best case with an open ring overlay network and four households. The number of sequential steps is $2|H|-1$. This results to 7 steps in case of four households and 99 in case of 50 households. If another overlay network is used, the amount of sequential steps decrease. In case of a star overlay network, only four sequential steps are required. This is independent on the number of households $|H|$ (see Fig. 9(b)). So the amount of sequential steps depends on the maximum number of hops. The small-world topology used for the overlay network here, has typically a logarithmically growing maximum number of hops. So the real best case in our scenario with the small-world overlay network has a number of sequential steps between 4 and 99. Please note that the convergence time has a linear dependency on the number of sequential steps.

However, simulations show that the worst case does not occur in practice. Convergence times of an exemplary day and a box plot of a one month simulation is shown in Fig. 10. The simulated best case is 3.8 s and the worst case 8.5 s. The median time for convergence is around 4.7 s.



(a) open ring (b) star

Fig. 9 Messages in the best case with four households and two different overlay networks

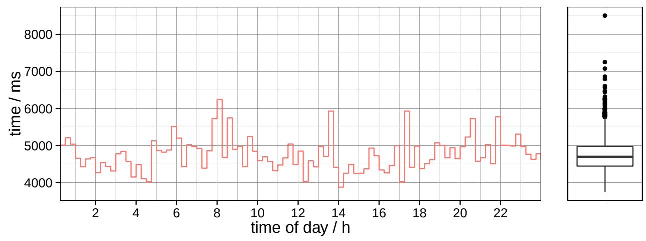


Fig. 10 Time for convergence (COHDA)

4.1.2 PowerMatcher

PowerMatcher always performs the same four communication steps to find a solution. As soon as the algorithm has performed these steps, it can be regarded as converged. The time for convergence is shown in Fig. 11. To prevent a concentrator overload, the households send their bids with an equal distributed time delay between 0 and 100 ms. This way the convergence time of PowerMatcher is almost a representation of this delay and an additive for the latency and time for transmission of around 215 ms in average.

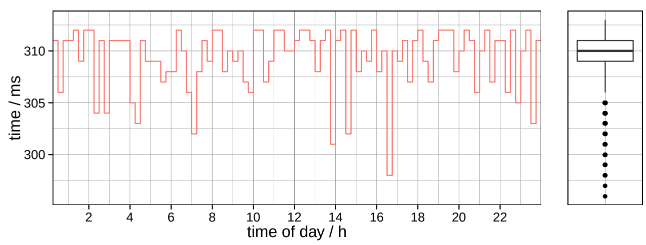


Fig. 11 Time for convergence (PowerMatcher)

4.1.3 PrivADE

PrivADE is a round-based algorithm. The communication is in principle organised unidirectional and no parallel communication steps occur. For this reason the converging times is proportional to the required number of rounds and the amount of households, which is 50. The first round needs 2.7 s and each additional round approx. 2.4 s. PrivADE needs two to six rounds in the simulated scenario. This results in convergence times from 5.1 s up to 15 s. Fig. 12 shows the convergence times of an exemplary day and a box plot for a simulation with a period of one month.

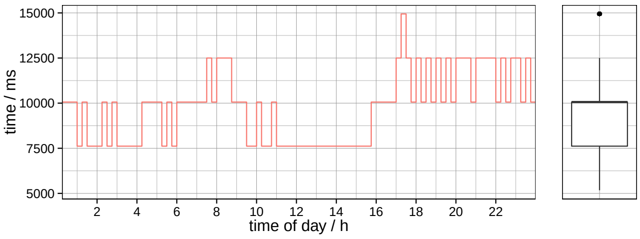


Fig. 12 Time for convergence (PrivADE)

4.1.4 Comparison of Convergence Times

The convergence times of COHDA and PrivADE algorithm are close to each other in this scenario (4.7 s compared to 9.7 s). PowerMatcher is much faster with times around 0.3 s. All convergence times are below 15 s, which enables an execution in a one minute interval.

4.2 Data Requirements

In this section, the data amount of the different EMAs is analysed. Simulation period is one month. The figures in this section illustrate a curve for an exemplary day as well as a box plot for one month, which corresponds to the entire simulation.

4.2.1 COHDA

Similar to COHDAs time for convergence, its best case for the amount of data can be shown at the example of Figure 9. With the open ring overlay network and four households the message amount is ten. Household 4, that calculates the final solution candidate, has to exchange 2 messages. Household 3 has to exchange 3, etc. For increasing number of households $|H|$ it results to

$$|M_{best,openring}| = 1 + \sum_{n=1}^{|H|-1} (n+1) = \frac{|H|^2 + |H|}{2} \ . \quad (1)$$

This results in a message amount of 1274 for 50 households. In case of the star overlay network, the number of messages is

$$|M_{best,star}| = 1 + 3|H| \ , \quad (2)$$

which leads to 151 messages for 50 households. The best case in the simulation is 801 messages and thereby between both (see Fig. 13).

The message size varies dependent on the number of households that are considered in the message ($|C|$). The message size can be calculated as follows: 38 Byte $+|C|\cdot$64 Byte. This results in a maximum message size of 3238 Byte for 50 households.

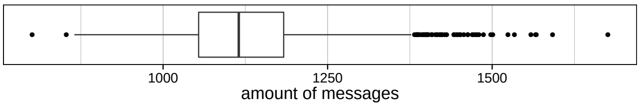


Fig. 13 Total message amount for convergence (COHDA)

The simulated amount of data needed to converge is shown in Fig. 14. This data amount varies between 600 kB and above 1600 kB. Fig. 15 shows that the total amount of received data varies considerably in the different households. Household 25 only receives 15.6 kB in average, compared to household 45 that receives 46.5 kB in average. Please note that the amount of transmitted data is basically equal to the received data in COHDA.

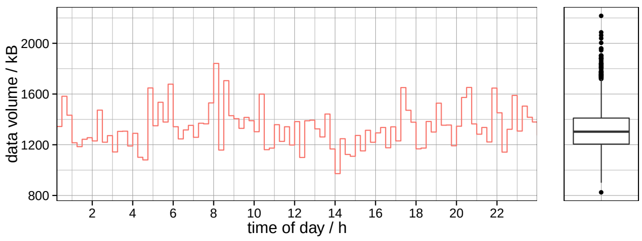


Fig. 14 Data volume needed for converging using the COHDA algorithm (including MAC headers)

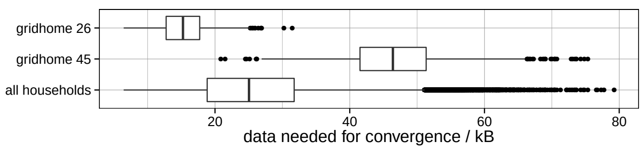


Fig. 15 Amount of received data per household required to converge using the COHDA algorithm

4.2.2 PowerMatcher

The more devices are manageable in a household, the more extensive is the bid-curve. This results in a higher data volume. The most complex household bid-curve-message has only 78 Byte, including the MAC header of 30 Byte. This corresponds to six stored coordinates. On the other hand, the smallest bid-message has a size of 38 Byte. In this case, only one tuple containing price and consumption has to be transmitted. If a household contains an adaptable device, at least one more tuple needs to be sent. While a μCHP only requires a single additional tuple, a battery storage requires three extra tuples. This is due to the more complex bid-curve. Fig. 16 shows the spread of bid-curve-message sizes. The sum of the total data volume sent by all participants is shown in Fig. 17. Depending on the number of controllable devices, the total data volume varies from 4.7 kB up to 5.6 kB. Especially during the evening when a lot of EVs are at home, a lot of devices are controllable.

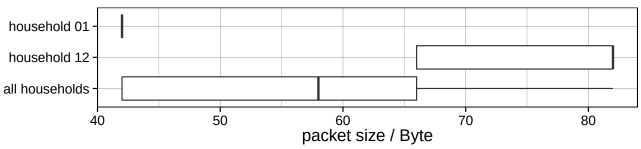


Fig. 16 Bid-curve-message size of all households and the households with the lowest and the highest average package size (including MAC headers)

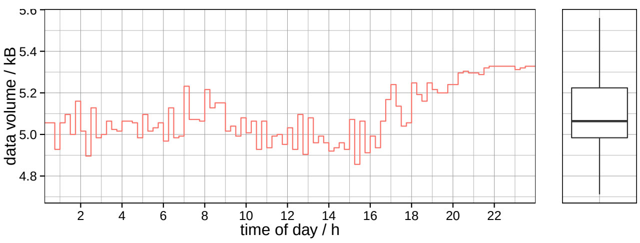


Fig. 17 Data volume needed for converging of PowerMatcher (including MAC headers)

4.2.3 PrivADE

Analysing the amount of data transferred for convergence using the PrivADE algorithm leads to a similar outcome as for the required convergence time (compare Fig. 12 and Fig. 18). This is due to the round-based approach of PrivADE. The first round requires approx. 120 kB of data. Second or later rounds only need approx. 8 kB. So the total data volume required by PrivADE varies between 128 and 161 kB.

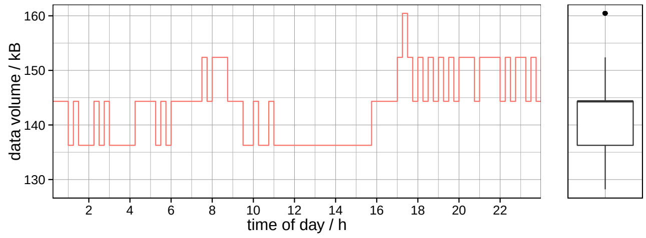


Fig. 18 Data volume needed for converging of PrivADE (including MAC headers)

4.2.4 Data Comparison

Similar to the time required for convergence, the amount of data depends on the selected algorithm as well. While PowerMatcher leads to very low data receptions and transmissions for households, the concentrator has to handle messages from each household in parallel. Using COHDA or PrivADE leads to lower communication requirements on the server side (including concentrator), because they are based on a more distributed approach. Therefore, households need to exchange more data. However, in the considered scenario, the average number of messages is 11 per household for COHDA and 4 for PrivADE, as well as their total data size around 1.5 MB or 144 kB. Both algorithms can be handled by most communication technologies. Table 3 gives an overview of the required communication.

Table 3 Average traffic needed for convergence of COHDA, PowerMatcher and PrivADE (including MAC headers)

		COHDA data/count	PowerMatcher data/count	PrivADE data/count
household	rx	26.4 kB/11	38 Byte/1	2805 Byte/4
	tx	26.4 kB/11	55.6 Byte/1	2805 Byte/4
server/ auctioneer	rx	-	374 Byte/1	2805 Byte/4
	tx	38 Byte/1	38 Byte/1	2805 Byte/4
concen- trator	rx	NA	2818 Byte/51	NA
	tx	NA	2274 Byte/51	NA

4.3 Scalability

In this section the scalability regarding increasing number of households of the different algorithms is analysed. Therefore, the required time and data for convergence is evaluated. The number of controllable devices per household remains constant.

4.3.1 COHDA

In small-world topologies, the average minimum path length l increases logarithmically with the number of nodes $|H|$ ($l \propto \log |H|$) [12]. The average number of neighbours remains constant at three. As a result, it can be assumed that the average time to reach convergence increases logarithmically with the number of households ($\mathcal{O}(\log |H|)$).

The total data volume for convergence increases much faster than the time for convergence. This is due to two additional scaling effects. Firstly, the number of messages $|M|$, that will be sent simultaneously, increases linearly with the number of households $|H|$, because all households send messages in parallel ($|M| \propto |H|$). Secondly, the average message size m_{size} increases linearly with $|H|$, because information about each household has to be communicated ($m_{size} \propto |H|$). In addition to the time effect, this results to a data amount scaling behaviour of $\mathcal{O}(|H|^2 \log |H|)$.

Simulations that are shown in Fig. 19 confirm both scaling assumptions. This leads to a total data transmission of more than 1 GB in case of 1000 households.

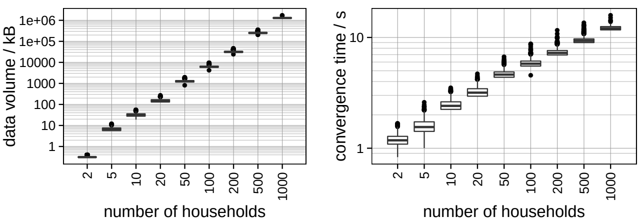


Fig. 19 Time and total data volume needed for convergence by varying the number of households (COHDA)

4.3.2 PowerMatcher

For the scaling analysis of PowerMatcher, the number of concentrators is fixed to one. Because of the four same steps in PowerMatcher, the algorithm scales independent from the number of households ($\mathcal{O}(1)$). Small effects on the scaling behaviour are due to the network topology: the maximum number of hops increases logarithmically till 80 nodes are reached (see section 3.3). The time of the slowest communication path in each of the four steps (Section 2.2.2) determines the total time. Due to the topology and the random time delay in each household, a slight increase can be expected. The amount of messages $|M|$ increases linearly with the number of households $|H|$ and the amount of concentrators $|C|$ ($|M| = 2|H| + 2|C|$). Due to the fact that there are much more households, than concentrators ($|H| >> N_C$), the increase of messages can be described as $\mathcal{O}(|H|)$. Because of an almost constant average message size, the data volume increases also linearly with the number of households ($\mathcal{O}(|H|)$).

Simulations with the amount of data and the required convergence time are shown in Fig. 20. The time for convergence increase slightly with the number of households. The average time increases from 226 ms in case of 2 households up to 340 ms in case of 1000 households. This behaviour is expected. The simulations show, that the data volume increases slightly less than linear. This can be explained with the aggregated bid, which is sent from the concentrator to the auctioneer. Its size increases less than linear because some prices in households bids-curves are the same. This is an economy of scale effect.

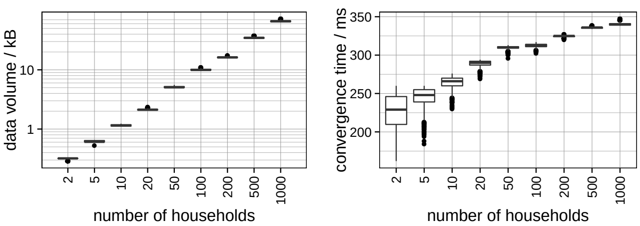


Fig. 20 Time and total data volume needed for convergence by varying the number of households (PowerMatcher)

4.3.3 PrivADE

In order to analyse the scaling behaviour of PrivADE, firstly the number of required rounds $|R|$ has to be considered. In Fig. 21, it can be seen that the number of rounds increases less than double logarithmic with the amount of households ($\mathcal{O}(\log \log |H|)$) in our scenarios.

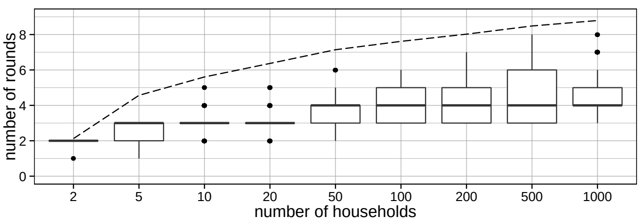


Fig. 21 Required rounds by varying the amount of households (PrivADE). The dashed line represents a curve with a $\log \log |H|$ behaviour.

Due to the fact that each round needs $|H| + 1$ communication steps, the convergence time increases linear with the number of rounds $|R|$ and the number of households $|H|$ ($\mathcal{O}(|H| \log \log |H|)$). This leads to convergence times up to 347 s in case of 1000 households. The data volume does not increase as fast as the time to converge, because the size of data exchanged in the second and later rounds is smaller than for the first round. The scaling behaviour of time and data required for convergence is shown in Fig. 22.

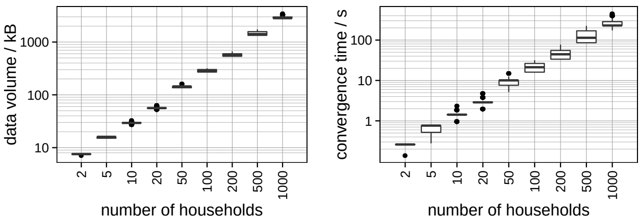


Fig. 22 Time and total data volume needed for convergence by varying the number of households (PrivADE)

4.3.4 Scalability Comparison

The scaling behaviour of the algorithms regarding increasing numbers of households is different. Two aspects have been analysed, the data amount and the time for convergence. Regarding data, PowerMatcher scales linear with the number of households. PrivADE is a bit worse and needs slightly more data with increasing households. COHDA, on the other side, needs much more data. It scales worse than quadratic with the number of households. Table 4 shows an overview of the scaling behaviour. Regarding time for convergence, PowerMatcher achieve the best results again. It converges nearly independent of the number of households. The convergence time of PrivADE increases slightly faster than linear and the time-scalability of COHDA is between PowerMatcher and PrivADE. Fig. 23 shows, that PrivADE and PowerMatcher converge equally fast in case of 2 households. Because of better scalability, Private become slower with increasing number of households. From 20 households upwards, the worse scaling of PrivADE allows COHDA to be second best.

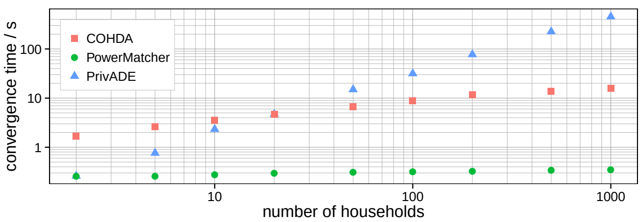


Fig. 23 Convergence times comparison by varying the number of households in our scenario

4.4 Communication Limitations

In this section, the convergence times of the algorithms are analysed by variation of available bandwidth and latency. Thereby, only bandwidth and latency of the leafs

Table 4 Scalability comparison of COHDA, PowerMatcher and PrivADE in our scenario

	total data amount	convergence time
COHDA	$\mathcal{O}(\lvert H\rvert^2 \log \lvert H\rvert)$	$\mathcal{O}(\log \lvert H\rvert)$
PowerMatcher	$\mathcal{O}(\lvert H\rvert)$	$\mathcal{O}(1)$
PrivADE	$< O(\lvert H\rvert \log\log \lvert H\rvert)$	$\sim O(\lvert H\rvert \log\log \lvert H\rvert)$

(see topology in Fig. 8) are limited. These leafs represent households, servers, concentrators and the auctioneer. The bandwidth and latencies between routers remain unaffected ($1\,\mathrm{Gbit\,s^{-1}}$ and $2\,\mathrm{ms}$).

4.4.1 COHDA

In the considered scenario the minimum data rate per household is $50\,\mathrm{kbit\,s^{-1}}$, when using the COHDA algorithm. In case of lower bandwidths, the algorithm does not converge reliably within the 15 minutes interval. This is independent of the latency. The convergence time in dependency of the data rate and the latency is shown in Fig. 24. High latencies become relevant at higher data rates. For example a latency of 200 ms, compared to 2 ms, slows down the convergence by approx. 7 s at high data rates.

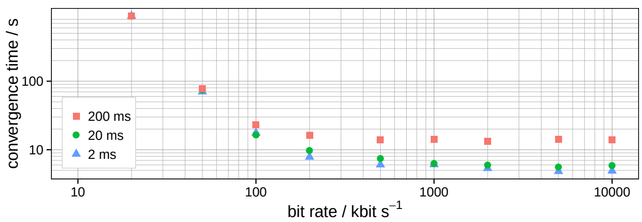


Fig. 24 Maximum time for convergence of COHDA by varying the bandwidth and the latency

4.4.2 PowerMatcher

In PowerMatcher, a latency of 200 ms slows down the convergence by 1.6 s. This is due to the four steps of PowerMatcher, where each step needs the sum of all component delays that are within the communication path. This corresponds to two times the latency of the leafs (200 ms) plus up to six times the latency of the routers (2 ms) at each of the four steps. These 1.6 s are negligible when considering that one minute is the finest interval that is taken into account.

Regarding low data rates, PowerMatcher shows fairly robust results too. Fig. 25 shows, that $100\,\mathrm{bit\,s^{-1}}$ are already sufficient to reach convergence in 500 s, which is well within the 15 minute interval.

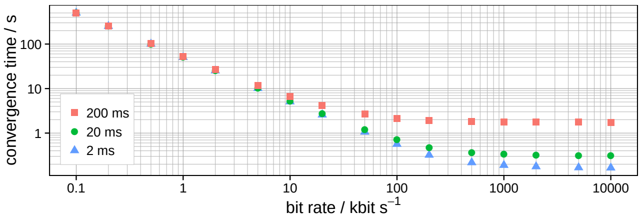


Fig. 25 Maximum time for convergence of PowerMatcher by varying the bandwidth and the latency

4.4.3 PrivADE

In contrast to COHDA and PowerMatcher, the influence of large delays of household d_h and the server d_s is very high on PrivADE because it requires $|H|+1$ sequential communication steps per round. This leads to a total convergence deceleration of $t_d = 2\cdot(d_s+\sum_{h\in H} d_h)$ alone through the leaf delay. This time is 20.4 s each round in case of a 200 ms leaf delay. In our scenario the maximum number of rounds is six, which leads to a total time delay of 122.4 s. Thus, an interval of one or two minutes is prohibited, even in case of very high bandwidths.

The bandwidth limitation cause a further time delay of $t_b = (|H|+1)\cdot\frac{5438\text{Byte}}{b}$. The 5438 Byte is the data amount, that have to be send sequential in six rounds.

In sum, the convergence time of PrivADE composing the addition of t_d, t_b and the time for transmission trough the higher layers of the physical topology, which is very low. In our scenario, PrivADE can be executed reliably with bandwidths down to 5 kbit s^{-1}.

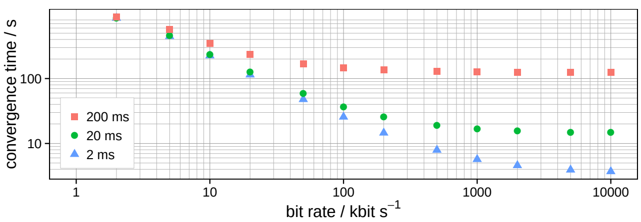


Fig. 26 Maximum time for convergence of PrivADE by varying the bandwidth and the latency

4.4.4 Comparison of Communication Limitations effects

The influence of communication limitations on the considered algorithms vary significantly. While COHDA is prone to bandwidth limitations, higher latencies does not have a large impact. PrivADE shows the exact opposite behaviour. A latency of 200 ms leads to a total convergence deceleration of approx. 100 s. A lower bandwidth on the other side has no great influence. In general the effect on PowerMatcher is not as high as the effect on both other algorithms. All in all, PowerMatcher can be used with bandwidths as low as 100 bit s^{-1}, PrivADE requires at least 2 kbit s^{-1} and COHDA a minimum of approx. 30 kbit s^{-1}.

4.5 Algorithm Comparison

The energy management, convergence times, data requirements, scalability and behaviour on communication limits have been analysed for COHDA, PowerMatcher and PrivADE. Regarding energy management, all algorithms have a similar ability to reduce consumption peaks or shape the load.

Regarding communication costs, PowerMatcher has the fewest requirements. It requires few data, so a limited bandwidth has low influence. Due to only four sequential communication steps, high latencies has limited influence too. Furthermore, PowerMatcher has the best scalability. However, in case of a shared communication medium, an individual time delay for each household should be considered. Otherwise, all households send messages at the same time. This leads to data collisions and can cause a temporary network overload. Due to the fact, that the concentrator is a part of each communication step, a performance upgrade of this node could improve the scalability characteristics even more. Furthermore, it is also possible to use more concentrators to split the load. However, PowerMatcher needs the auctioneer as a central unit and at least one concentrator.

COHDA needs the server only for an initial information about the goal consumption. The households then find a solution totally distributed. This is advantageous because no infrastructure has to be provided by the energy manager. In COHDA, each household is only aware of its own objective. The objectives of the other households are unknown, so many messages have to be transmitted to find a good solution. This leads to high parallel communication requirements and moderate scalability. However, for a limited number of households and communication technologies with the ability to handle a high data volume in parallel (e.g. DSL), COHDA can be well suited. Another advantage of COHDA is, that a convergence is possible, even if messages are lost or a node failure occur.

COHDA and PowerMatcher have in common, that parts of private data is disclosed. To preserve privacy fully, it is necessary that no participant knows any consumption values of any other household. This is the strength of PrivADE. It is using homomorphic encryption and is based on rounds. In this way only the server holds values of the households. However, these values are aggregated and no information about an individual household can be gained. Due to the round-based approach a large amount of sequential communication steps are necessary. On the one hand, this leads to convergence times which are strongly dependent on the latency of each household. On the other hand, there are no parallel communication steps, which limits the load of the total network. Therefore, a use of PrivADE is suitable for technologies with a shared medium like wireless communication or PLC.

Table 5 shows an overview of recommendations for the different algorithms.

Table 5 Recommendations for COHDA, PowerMatcher and PrivADE

COHDA	needs a network that enables high parallel communication, insensitive to high latencies, robust against node failures, moderate scalability, server only necessary for initiation
Power-Matcher	low bandwidth and latency requirements to the communication network, fast convergence, good scalability, auctioneer and at least one concentrator necessary
PrivADE	requires communication network with low latencies, good for shared medium technologies, moderate scalability, privacy preserving, one server necessary

5 Conclusion

Energy Management will become more and more important in the future. All three analysed algorithms are highly suitable to solve the emerging problems of our scenarios. However, the requirements on the underlying communication system vary significantly. If a high parallel communication network is available and a central unit is undesirable, COHDA can be recommended. If only a technology without ability of parallel communication is available and privacy is a concern, PrivADE is the best solution. However, PrivADE requires a communication technology without high latencies. If only fast convergence is required and a central unit is feasible, PowerMatcher will be the best choice. PowerMatcher has a good scalability and thus can handle very high number of households. This only requires an appropriate number of concentrators.

In future work scenarios with other communication technologies like PLC or mobile communication networks will be analysed. Furthermore, the effects of packet loss and node failures will be evaluated.

References

1. Brettschneider, D., Toenjes, R., Roer, P., Hoelker, D.: Distributed algorithm for energy management in smart grids. In: WTC 2014, pp. 1–6 (2014)
2. Conejo, A.J., Morales, J.M., Baringo, L.: Real-time demand response model. IEEE Transactions on Smart Grid **1**(3), 236–242 (2010). DOI 10.1109/TSG.2010.2078843
3. Hinrichs, C., Lehnhoff, S., Sonnenschein, M.: Agents and Artificial Intelligence: 5th International Conference, ICAART 2013, Barcelona, Spain, February 15-18, 2013. Revised Selected Papers, chap. COHDA: A Combinatorial Optimization Heuristic for Distributed Agents, pp. 23–39. Springer Berlin Heidelberg, Berlin, Heidelberg (2014). DOI 10.1007/978-3-662-44440-5_2. URL `http://dx.doi.org/10.1007/978-3-662-44440-5_2`
4. Hölker, D., Brettschneider, D., Fischer, M., Tönjes, R., Roer, P.: Quality-functions for an uniform and comparable analysis of demand side management algorithms. Computer Science - Research and Development pp. 1–8 (2014). DOI 10.1007/s00450-014-0280-x. URL `http://dx.doi.org/10.1007/s00450-014-0280-x`
5. Kok, J.K., Warmer, C.J., Kamphuis, I.: Powermatcher: multiagent control in the electricity infrastructure. In: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems, pp. 75–82. ACM (2005)
6. Saad, W., Han, Z., Poor, H.V., Basar, T.: Game-theoretic methods for the smart grid: An overview of microgrid systems, demand-side management, and smart grid communications. IEEE Signal Processing Magazine **29**(5), 86–105 (2012). DOI 10.1109/MSP.2012.2186410
7. Samadi, P., Mohsenian-Rad, H., Schober, R., Wong, V.W.S.: Advanced demand side management for the future smart grid using mechanism design. IEEE Transactions on Smart Grid **3**(3), 1170–1180 (2012). DOI 10.1109/TSG.2012.2203341
8. Statistisches Bundesamt: Statistisches Bundesamt, Ausstattung privater Haushalte mit ausgewählten Gebrauchsgütern (2013)
9. University of Applied Sciences Osnabrueck: Siena - simulating energy network applications. URL `http://siena.hs-osnabrueck.de/`
10. US Department of Energy: Communications requirements of smart grid technologies (2010)
11. Verein Deutscher Ingeniuere e.V.: VDI 4655:2008-05 Reference load profiles of single-family and multi-family houses for the use of CHP systems (2008)
12. Watts, D.J., Strogatz, S.H.: Collective dynamics of small-world networks. nature **393**(6684), 440–442 (1998)
13. Yan, Y., Qian, Y., Sharif, H., Tipper, D.: A survey on smart grid communication infrastructures: Motivations, requirements and challenges. IEEE Communications Surveys Tutorials **15**(1), 5–20 (2013)