

# A Game Theoretic Software Framework for Optimizing Demand Response

Wolfgang Lausenhammer and Dominik Engel

Josef Ressel Center for User-Centric Smart Grid Privacy,  
Security and Control  
Salzburg University of Applied Sciences  
Salzburg, Austria  
{wolfgang.lausenhammer,dominik.engel}@en-trust.at

Robert Green

Dept. of Computer Science  
Bowling Green State University  
Bowling Green, OH 43403  
greenr@bgsu.edu

**Abstract**—Demand response (DR) is a crucial and necessary aspect of the smart grid, particularly when considering the optimization of both, power consumption and generation. While many benefits of DR are currently under study, an issue of particular concern is optimizing end-users' power consumption profiles at various levels. This study proposes a fundamental, game theoretic software framework for DR simulation that is capable of investigating the effect of optimizing multiple electric appliances by utilizing game theoretic algorithms. Initial results show that by shifting the switch-on time of three household appliances provides a savings of up to 6%.

## I. INTRODUCTION

With the increasing pervasiveness of renewable energy new challenges have arisen: Energy is no longer exclusively produced in large power plants, but also in the homes of ordinary people. Eventually, this development will lead to a paradigm shift, away from the hierarchical top-down oriented system to a decentralized structure with volatile renewable energy sources, such as wind turbines, photovoltaic cells and plug in electric vehicles (PEV) [1,2].

By coordinating household appliances and PEVs, off-peak usage could result in cheaper electricity prices. With respect to coordination, demand response (DR) management could pose an ideal solution to this problem [3-5].

Within the vast amount of different approaches to simulate and model DR, game theory proves to be a capable method of modeling and describing complex interactions between different rational players. The goal of a game theoretic approach in DR management is to develop a model and proof that if every agent tries to maximize its own profit, an equilibrium point is found. By acting selfishly, players reach a global optimum [6]. Publications in this area range from load shifting approaches [7,8] to using storage devices such as PEVs in micro-grid storage games [9] to games that focus on utility companies [10,11].

This study proposes a new software framework – *Okeanos*<sup>1</sup> – that enables the simulation and study of these issues through the provision of an extensible, open source simulation platform that can both, model different types of loads and be configured

to work with different game theoretic DR management approaches. By providing a very lightweight interface for users to plug in their own control algorithms, the framework also allows for new strategies to be tested.

The remainder of the paper is structured as follows: An overview of related work in game theory in DR management, and software frameworks for DR management is presented in Section II; This is followed by introducing the novel DR simulation platform, *Okeanos*, and highlighting its key concepts in Section III; Initial results of using *Okeanos* are described in Section IV; and, finally, Section V concludes this work.

## II. LITERATURE REVIEW

This section provides an overview of the state-of-the-art in game theory in DR management and software frameworks for DR management.

### A. Game Theory in Demand Response Management

Game theory, in its essence, aims to help understand situations in which several decision-makers interact. Being a mathematical framework and analytical tool, game theory helps study the relationships and actions among rational players [6,12].

Saad and co-authors evaluate the available approaches for applying game theory to timely open and relevant smart grid related problems in [6]. They focus on three emerging areas, particularly: micro-grid systems, demand-side management, and communications.

One way to reduce the *peak-to-average-ratio* (PAR) of an energy system is to change the schedule of shiftable household appliances. Traditionally, multi-objective functions [13] and non-linear models [14,15] are used to determine a (near) optimal, (near) real-time schedule.

In contrast to that, Mohsenian-Rad and co-authors utilize game theory and propose an energy consumption game to optimize energy costs in [7]. Their aim is to change the daily schedule of shiftable household appliances. Although the schedule could be calculated centrally, calculation is done in a decentralized way. This is the preferable way, as it requires significantly less communication effort and does not provide a single point of failure [7].

<sup>1</sup> The project is released as open source and can be accessed at <https://github.com/wolfgang-lausenhammer/Okeanos>

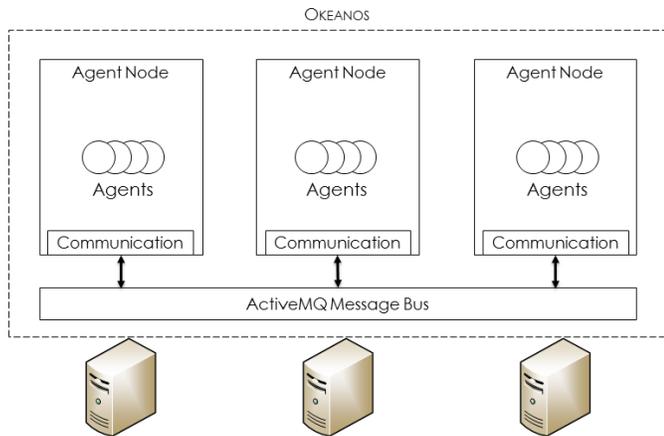


Fig. 1. Physical mapping of agents in Okeanos.

Unlike the aforementioned load shifting approach, the authors of [9] propose a non-cooperative micro-energy-storage game. Here, users decide on a storage profile for their household devices to optimize a utility function reflecting the cost. That is, users decide on a point in time when they want to buy energy, ideally during low-cost periods, and when they want to use this energy to satisfy their demand. Several constraints, such as the maximum capacity, the storage efficiency and running costs are considered [9].

### B. Software Frameworks for Demand Response

Recent [16-20], as well as older publications [21-25], propose a multi-agent approach as appropriate to deal with the complex topic of demand-response optimization.

*PowerMatcher* is a tool used for coordinating a large cluster of distributed energy resource devices within a smart grid in near real-time. Its focus is on end-consumers allowing them to use their appliances to actively participate in the energy market. Thus, by offering flexibility to the grid, customers get the possibility to reduce their energy bills. Appliances, represented by agents, coordinate consumption and production and calculate the market clearing price [16].

Similarly, in DEZENT, agents communicate their demand or supply to a balancing group manager (BGM) at their respective grid hierarchy level. It is the BGMs duty to match similar offers for demand and supply. Unmatched offers are handed over to the next higher level, where this matching starts over again [26].

*Okeanos* is fundamentally different to these approaches, as it plans consumption and production ahead of time and uses mathematically proved solutions for finding the optimal schedule for household appliances. Indeed, by using game theoretic approaches, it is guaranteed that if every user acts selfishly and optimizes his or her own costs, a global cost-optimum is established.

## III. OKEANOS – A MULTI-AGENT GAME THEORETIC DEMAND RESPONSE MANAGEMENT SOFTWARE FRAMEWORK

*Okeanos* is a novel Java-based multi-agent DR simulation platform with special focus on the compatibility to game theory. That is, not only one particular coordination mechanism

as in [16], [17] or [26] is supported, but any mechanism, as long as it complies with the specified interface.

The goal is to allow for a holistic approach to demand response management with a very extensible platform that can host all kind of appliances, as long as there is an appropriate driver available in the system. By defining a clear interface and basing the framework on OSGi, these drivers can be easily developed, deployed and removed from the system.

### A. Household appliances as the smallest active unit

*Okeanos* combines several features of [16], [17] and [26]. That is, *Okeanos* utilizes the multi-agent paradigm to represent household appliances. Thus, every single appliance within a household taking over an active role in DR management is represented by an agent. This implies that every agent can decide on its own and can pursue a target. The target is specified by the currently plugged in game. Likewise, the capabilities are specified by the underlying household device and the corresponding driver, which are both explained later on in this section.

To be able to focus on other aspects of the system, a feature-rich, modularized and easy to use framework is utilized for providing multi-agent features. A comparison between the Java-based multi-agent simulation platforms JADE, *Janus*, *Jason* and JIAC resulted in JIAC as the winner. Criteria included functionality, active development, ease of use and adoption throughout the software developer community.

JIAC's modern approach to use the *Spring* framework as the basis for the whole system is unique throughout all compared multi-agent frameworks. Additionally, the utilization of this framework assures the system to be future-proof according to the best of the authors' knowledge, thus making it first choice for implementing multi-agent systems in Java.

*Okeanos*, a JIAC-based multi-agent system, is structured as shown in Fig. 2. That is, the application can consist of several agent nodes, agents and agent beans [27].

Agent nodes are distributed containers providing the necessary infrastructure for agents, such as a communication infrastructure or white and yellow pages services [27].

Several agents, that is, household appliances, can be hosted within one agent node (see Fig. 1) and according to the *service oriented architecture* (SOA) architectural pattern, provide services to other agents. Moreover, as required by [27], every agent comprises several agent beans, which provide the actual functionality like persistent memory and usage of infrastructure services for inter-agent communication. As a consequence, functionality defined by *Okeanos*, such as weather service, pricing service or time service, are implemented as agent beans and OSGi bundles and plugged into the agents as such.

### B. Plug in support

To be able to plug in different device drivers or games, flexible and powerful interfaces need to be developed. Furthermore, implementations of these interfaces need to be hooked into the system easily, in order to keep the threshold for developing modules as low as possible. For that reason, it is advantageous to modularize the system as much as possible.

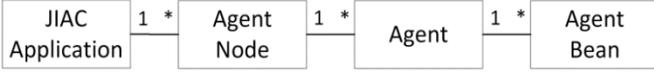


Fig. 2. Structure of a JIAC-based multi-agent system. Adapted from [27].

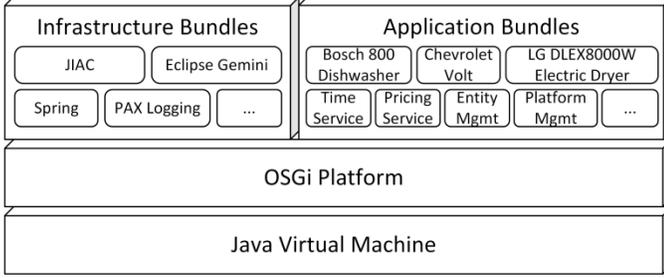


Fig. 3. Okeanos Bundle structure with sample household devices and services

With OSGi designed as a *service oriented architecture*, *Okeanos* features no monolithic core, but is a conglomerate of various bundles (see Fig. 3). According to the OSGi R5 specification [28], it is best practice to keep the interfaces in a separate bundle, to also allow for optional bundles not being present in the OSGi container. Consider, for example, a logging service: The application does not necessarily need an implementation for a correct execution, however, at least the interface needs to be present, otherwise OSGi would not be able to resolve the dependencies of the bundle.

As indicated by Fig. 3, every service in *Okeanos* is represented by its own module and, therefore, separated in its own bundle. Their respective interfaces are all consolidated in interface bundles corresponding to the actual layer the bundle is part of. Likewise, as it is possible to have no implementation present in an OSGi container, it is possible to have multiple implementations present. This is especially true for device drivers, as they all implement the same interface. Therefore, the service user needs to select from the list of available drivers.

The service provider, that is the driver, can specify additional properties, such as year and brand of a household device, as key value pairs to supply the service user with some cues.

### C. Game theory in *Okeanos*

As described in Section II, game theory can be used to understand the result of the dynamism in a game between several interacting players. Every player in such a game is represented by its own agent in *Okeanos*. By that, the prerequisite that players have to act rationally can be assured.

There are a number of published game theoretic approaches to DR management [7-11]. Some take load shifting into consideration, some make use of available storage devices such as PEVs, and others formulate a game with multiple utility companies.

*Okeanos* is designed to support any game that can be mapped to the specified interface. Therefore, it is crucial to define the interface as general as possible, while at the same time being specific enough that implementations of the interface have a useful basis for doing their optimization and calculation.

Furthermore, it is possible that individual agents use different games. The meaningfulness of such a mixture is, however, questionable, as no guarantee of the existence of a Nash equilibrium can be given under such circumstances.

As a first proof of concept, the game proposed by Mohsenian-Rad and co-authors [7] has been modelled with *Okeanos*. One of the reasons for this is that the authors formulate their algorithm in pseudo code, which allows for accurate adaptation. Moreover, by utilizing load shifting, potentially more devices can be integrated in the first place as it were possible with storage devices.

The decentralized objective function of the game in [7] is given by (1) where  $x_{n,a}^h$  represents a one hour energy consumption scheduled for appliance  $a$  of user  $n$  at hour  $h$ . Additionally, the cost functions  $C_h$  are increasing and strictly convex.  $\mathcal{H}$  is the set of possible hours of the 24h time horizon [7].

$$\begin{aligned} & \underset{x_{\{n\}} \in \mathcal{X}_n}{\text{minimize}} \sum_{h=1}^H C_h \left( \sum_{a \in \mathcal{A}_n} x_{n,a}^h + \sum_{m \in \mathcal{N} \setminus \{n\}} l_m^h \right) \\ & \text{subject to } l_m^h = \sum_{a \in \mathcal{A}_n} x_{m,a}^h \quad h \in \mathcal{H} \end{aligned} \quad (1)$$

The consumption of all other players  $l_m^h, m \in \mathcal{N} \setminus \{n\}$  is static, therefore, only the schedule  $x_{n,a}^h$  of the local appliances  $a$  of player  $n$  at each hour  $h$  needs to be calculated. That is, an optimal schedule with respect to the consumption patterns of the other players needs to be computed [7].

The algorithm to play this game is given in Algorithm 1. The initial consumption is initialized randomly, because the game guarantees to find the Nash equilibrium regardless of the initial configuration. After that, every appliance finds the best solution to the local optimization problem (1) at random instances, e.g., by using the Interior Point Method (IPM). This randomness is important to allow for another appliance being faster with finding a solution and sending an update of its consumption. If a different solution to the optimal consumption is found, it is broadcast to the other devices. This loop is repeated until no schedules are changed anymore.

```

Randomly initialize  $l_n$  and  $l_{-n}$ 
repeat
  at random time instances do
    Solve local problem (1) using IPM.
    if  $x_n$  changes compared to current schedule then
      Update  $x_n$  according to the new solution.
      Broadcast a control message to announce  $l_n$  to the other agents across the system.
    end
  end
  if a control message is received then
    Update  $l_{-n}$  accordingly.
  end
until no agent announces any new schedule

```

Algorithm 1: Energy consumption game, executed by each user  $n \in \mathcal{N}$  [7].

Mohsenian-Rad and co-authors use an hourly consumption schedule for all devices. In contrast, *Okeanos* uses a 15 minutes interval to allow for a more fine-grained control over faster devices.

#### IV. INITIAL RESULTS

*Okeanos* is evaluated to test its applicability to real world problems and use cases. Therefore, this section gives an initial insight into the capabilities of *Okeanos*. Test results are based on the devices listed in Table 1 in addition to a household load profile. The data for implementing drivers for clothes washer, clothes dryer and a dishwasher is taken from [29]. The household data is based on the H0 load profile provided by the Bundesverband der Energie- und Wasserwirtschaft (Federal Association of the Energy and Water Industry) [30]. The H0 load profile is a standardized profile used to approximate the consumption of customers that cannot be measured otherwise.

The real-time pricing costs are taken from [31]. In order to draw a sound conclusion, the consecutively mentioned experiments were repeated at least 100 times and the reported results are average values. A single household with a 30 kWh load profile is used as a base case.

Starting with multiple devices within one household, the interaction between the devices is tested. The devices search for the point in time which minimizes the electricity costs for that device. The impact of shifting the load profile of a household is depicted in Fig. 4. Devices in the first chart run daily, whereas devices in the second run every third day. Additionally, to make the simulation more realistic and to take the consumption patterns of different households into account, the H0 load profile is shifted 0,  $\pm 1$ h or  $\pm 3$ h.

The major result of this simulation is that the more the regular households differ in their consumption patterns, the more the total load curve evens out. With all households using the standardized H0 load profile, several peaks are present, most notably those at 1 p.m. and 8 p.m. Naturally, considering the price per kWh, it is preferable to, especially at those hours, to reduce the energy consumption.

The only difference between the charts in Fig. 4 is the peak in the morning, when all the load shifting devices are switched on. This difference is due to the fact that the devices run only every third day and, therefore, on average, the consumption at that point should only be one third of that when they are switched on every day.

It can be seen in Table 2 that the effect of varying the load profile of households is negligible. This is valid throughout all compared categories.

Table 1: Overview of drivers used for evaluation. Data from [29] and [30].

Appliance	Model	Rating
Household	Standard load profile	Scaled to 30kWh
Clothes washer	LG WM2016CW	120V, 60Hz, 5A
Clothes dryer	LG DLE2516W	120V, 60Hz, 26A
Dishwasher	Kenmore 665.13242K900	120V, 60Hz, 9.6A

Table 2: Comparison of costs per month per household with load shifting in relation to shifted household load profiles.

	Regular 30kWh household	28kWh household with 2kWh load shifted devices	
		Run daily	Run every third day
0h shifting	\$85.80	\$82.25 (4,14%)	\$80.71 (5,93%)
$\pm 1$ h shifting	\$85.72	\$82.17 (4,14%)	\$80.66 (5,90%)
$\pm 3$ h shifting	\$85.10	\$81.60 (4,11%)	\$80.11 (5,86%)

Actual savings, according to the outcomes (see Table 2), can be noticed between a regular 30kWh household and when load shifting is in place. The average savings is around 4.14% if load shifting is in place.

Naturally, the savings of a household with its devices running only every third day needs to be higher compared to a household where the devices run every day. The savings compared to a regular household with no load shifting are 5.9%.

#### V. CONCLUSION

In this paper, we proposed *Okeanos*, a novel multi-agent demand response simulation platform that is capable of evaluating game theoretic approaches. Due to its extensibility, *Okeanos* can support a wide range of different household appliances. Moreover, because the system is based on OSGi, exchanging specific implementations is very easy, as long as it implements the same interfaces.

Initial results show that by optimizing three household appliances of one household, *Okeanos* can save up to 5.9% of energy costs per month. Future work will focus on studying the impact of more households, as well as integrating plug in electric vehicles in the simulation.

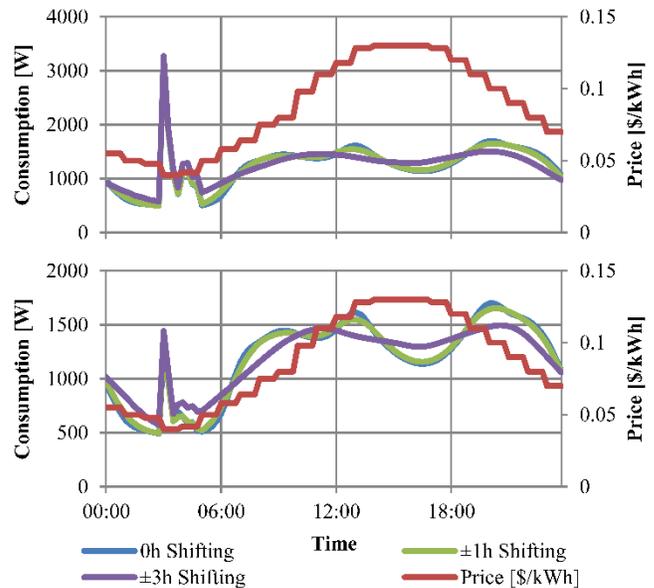


Fig. 4. Optimizing the schedule of one 28kWh/day household. Devices run every day in the first chart, every third day in the second chart.

## REFERENCES

- [1] P. Palensky and D. Dietrich, "Demand side management: demand response, intelligent energy systems, and smart loads," *IEEE Trans. Ind. Informat.*, vol. 7, no. 3, pp. 381-388, 2011.
- [2] J. Schwarzer, A. Kiefel and D. Engel, "The role of user interaction and acceptance in a cloud-based demand response model," in *39th Annu. Conf. IEEE Industrial Electronics Society (IECON 2013)*, Vienna, Austria, 2013, pp. 4797-4802.
- [3] Z. Fan, P. Kulkarni, S. Gormus, C. Efthymiou, G. a. S. M. a. Z. Z. Kalogridis, S. Lambotharan and W. Chin, "Smart grid communications: Overview of research challenges, solutions, and standardization activities," *IEEE Commun. Surveys & Tutorials*, vol. 15, no. 1, pp. 21-38, 2012.
- [4] A. Patel, J. Aparicio, N. Tas, M. Loiacono and J. Rosca, "Assessing communications technology options for smart grid applications," in *2011 IEEE Int. Conf. Smart Grid Communications (SmartGridComm 2011)*, Brussels, Belgium, 2011, pp. 126-131.
- [5] V. Murthy Balijepalli, V. Pradhan, S. Khaparde and R. Shereef, "Review of demand response under smart grid paradigm," in *2011 IEEE PES Innovative Smart Grid Technologies-India (ISGT 2011)*, Kollam, Kerala, India, 2011, pp. 236-243.
- [6] W. Saad, Z. Han, H. V. Poor and T. Başar, "Game-theoretic methods for the smart grid: an overview of microgrid systems, demand-side management, and smart grid communications," *IEEE Signal Processing Mag.*, vol. 29, no. 5, pp. 86-105, 2012.
- [7] A. Mohsenian-Rad, V. W. Wong, J. Jatskevich, R. Schober and A. Leon-Garcia, "Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid," *IEEE Trans. Smart Grid*, vol. 1, no. 3, pp. 320-331, 2010.
- [8] C. Ibars, M. Navarro and L. Giupponi, "Distributed demand management in smart grid with a congestion game," in *2010 1st IEEE Int. Conf. Smart Grid Commun. (SmartGridComm 2010)*, Gaithersburg, MD, 2010, pp. 495-500.
- [9] P. Vytelingum, T. D. Voice, S. D. Ramchurn, A. Rogers and N. R. Jennings, "Agent-based micro-storage management for the smart grid," in *Proc. 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, Toronto, Canada, 2010, pp. 39-46.
- [10] S. Maharjan, Q. Zhu, Y. Zhang, S. Gjessing and T. Başar, "Dependable demand response management in the smart grid: a Stackelberg game approach," *IEEE Trans. Smart Grid*, vol. 4, no. 1, pp. 120-132, 2013.
- [11] S. Bu, F. R. Yu and P. X. Liu, "A game-theoretical decision-making scheme for electricity retailers in the smart grid with demand-side management," in *2011 IEEE Int. Conf. Smart Grid Commun. (SmartGridComm 2011)*, Brussels, Belgium, 2011, pp. 387-391.
- [12] C. Rieck, *Spieltheorie; Eine Einführung*, 11th ed (in German). Eschborn, Germany: Christian Rieck Verlag, 2012.
- [13] M. A. A. Pedrasa, T. D. Spooner and I. F. MacGill, "Scheduling of demand side resources using binary particle swarm optimization," *IEEE Trans. Power Syst.*, vol. 24, no. 3, pp. 1173-1181, 2009.
- [14] P. Faria, Z. Vale, J. Soares and J. Ferreira, "Demand response management in power systems using a particle swarm optimization approach," *IEEE Intell. Syst.*, vol. 28, no. 4, pp. 45-51, 2011.
- [15] N. Gudi, L. Wang, V. Devabhaktuni and S. S. S. R. Depuru, "Demand response simulation implementing heuristic optimization for home energy management," in *North American Power Symposium (NAPS 2010)*, 2010, Arlington, TX, 2010, pp. 1-6.
- [16] K. Kok, "The PowerMatcher: Smart Coordination for the Smart Electricity Grid," Ph.D. dissertation, Free Univ. Amsterdam, Amsterdam, Netherlands, 2013.
- [17] O. Struß, "Open-Source-Modellierung und auktionsorientierte Regulierung dezentraler Energienetze," (in German), diploma thesis, Dept. Computer Science and Mathematics, Univ. Bremen, Bremen, Germany, 2012.
- [18] S. Rohjans, S. Lehnhoff, S. Schütte, S. Scherfke and S. Hussain, "mosaic — A modular platform for the evaluation of agent-based Smart Grid control," in *4th European Innovative Smart Grid Technologies (ISGT 2013)*, Lyngby, Denmark, 2013, pp. 1-5.
- [19] S. Lehnhoff, *Dezentrales vernetztes Energiemanagement: ein Ansatz auf Basis eines verteilten adaptiven Realzeit-Multiagentensystems* (In German). Heidelberg, Germany: Springer DE, 2010.
- [20] K. Kok, G. Venekamp and P. Maccougall, "Market-based control in decentralized electrical power systems," in *1st Int. Workshop on Agent Technologies for Energy Systems (ATES 2010)*, Toronto, Canada, 2010, pp. 61-66.
- [21] R. Gustavsson, "Agents with power," *Commun. of the ACM*, vol. 2, no. 3, pp. 41-47, 1999.
- [22] P. Carlsson, "Algorithms for electronic power markets," Ph.D. dissertation, Uppsala Univ., Uppsala, Sweden, 2004.
- [23] F. Ygge and H. Akkermans, "Power load management as a computational market," in *2nd Int. Conf. Multi-Agent Systems (ICMAS 1996)*, Menlo Park, CA, 1997, pp. 393-400.
- [24] K. Kok, Z. Derzsi, M. Hommelberg, C. Warmer, R. Kamphuis and H. Akkermans, "Agent-based electricity balancing with distributed energy resources, a multiperspective case study," in *Proc. 41st Annu. Hawaii Int. Conf. on System Sciences (HICSS 2008)*, Waikoloa, HI, 2008.
- [25] K. Kok, C. Warmer and I. Kamphuis, "PowerMatcher: multiagent control in the electricity infrastructure," in *Proc. 4th Int. Joint Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2005)*, Utrecht, Netherlands, 2005, pp. 75-82.
- [26] S. Lehnhoff, O. Krause, C. Rehtanz and H. Wedde, "Distributed autonomous power management-For a reliable operation under feasibility constraints," *Automatisierungstechnik*, vol. 59, no. 3, pp. 167-179, 2011.
- [27] JIAC Development Team, "JIAC - Java Intelligent Agent Componentware," DAI-Labor, Tech. Univ. Berlin, Berlin, Germany, Manual. Version 5.1.3, 2012.
- [28] *OSGi Core Release 5*, OSGi Alliance, San Ramon, CA, 2012.
- [29] Pipattanasomporn, M., Kuzlu, M., Rahman, S., and Teklu, Y., "Load profiles of selected major household appliances and their demand response opportunities," *IEEE Trans. Smart Grid*, vol. 5, no. 2, pp. 742-750, 2014.
- [30] Bundesverband der Energie- und Wasserwirtschaft, "Standardlastprofil H0: Haushalt, Privatverbrauch, gegebenenfalls geringfügig gewerblicher Bedarf", (in German), 2014.
- [31] Office of the Ohio Consumers' Counsel, "Smart grid: dynamic and time-of-use pricing", Office of the Ohio Consumers' Counsel, Columbus, OH, Tech. Rep., 2011.