Overview of machine learning and data-driven methods in agent-based modeling of energy markets

Ashreeta Prasanna [1], Sascha Holzhauer[2] and Friedrich Krebs[3]

Abstract: Local energy markets (LEM) allow prosumers and consumers to trade energy directly between one another and offer flexibility services to the grid. The benefits and challenges of such LEM need to be identified, and agent-based modeling (ABM) is a useful method to conduct simulation experiments that compare different market structures and clearing mechanisms. Machine learning (ML) and data-driven methods when integrated with ABM show great potential for constructing new distributed, agent-level knowledge. In this paper, we discuss the requirements for coupling ML methods and ABM. We also provide an overview of published literature on the common methods of integration of ML and data-driven methods in ABM and discuss how these requirements are commonly addressed.

Keywords: machine learning, agent-based modeling, local energy markets, reinforcement learning, load forecasting

1 Introduction

The widespread adoption of renewable energy supply technologies and the availability of data from smart meters and other monitoring systems allows the development of local energy markets (LEM), also referred to as peer-to-peer markets or direct energy markets. LEM aim to offer multiple benefits: implementation of prosumers' preferences: for instance for renewable energy or lower CO_2 emissions; reduction in energy costs; reduction in costs for grid investment; and flexible and efficient locally managed energy supply [Fa14, So18]. Price signals that indicate scarcity and excess of fluctuating energy supply could incentivize prosumers to act beneficially for the energy system. A number of projects use demonstration and/or modeling to analyze the benefits and drawbacks of LEM and their design [BOR17, Mo18, Me18, RKF16, RM13, So18, Zh18]. ABM is found to be particularly suitable to evaluate the design of LEMs because it allows the representation of aspects such as learning effects in repeated interactions, asymmetric information, imperfect competition, or strategic interaction and collusion in a more realistic way [RKF16, Se07].

[1] University of Kassel, Department Integrated Energy Systems, Wilhelmshöher Allee 73, 34121 Kassel, Germany, ashreeta.prasanna@uni-kassel.de
[2] University of Kassel, Department Integrated Energy Systems, Wilhelmshöher Allee 73, 34121 Kassel, Germany, sascha.holzhauer@uni-kassel.de
[3] University of Kassel, Department Integrated Energy Systems, Wilhelmshöher Allee 73, 34121 Kassel, Germany, friedrich.krebs@uni-kassel.de

doi:10.18420/inf2019_73

As agent-based models (ABM) are data intensive, automating or semi-automating the process of capturing system knowledge using ML and other data-driven methods is a growing field of research. This is especially true for LEM when agents interact with the dynamic energy system and time constraints need to be considered in forecasting market prices, energy consumption and generation as well in the bidding process.

ML algorithms can be classified into three broad categories: supervised learning, unsupervised learning and reinforcement learning [Al10]. Supervised learning algorithms are used to develop a predictive model based on both input and output data. Some examples of supervised learning algorithms are k-Nearest neighbors, support vector machines, decision trees, neural networks, etc. Unsupervised learning algorithms are used to group and interpret data based only on input data. Common unsupervised learning algorithms are k-means clustering, hierarchical clustering, DBSCAN (Density-Based Spatial Clustering of Applications with Noise) clustering, etc. Reinforcement learning (RL) is a goal directed approach where an agent learns the optimal behavior through repeated trial-and-error interactions with the environment without human involvement. Examples of RL algorithms are Q-learning, genetic algorithms, Erev-Roth reinforcement learning, learning classifier systems, etc.

ML can be coupled with ABM in a number of different ways (see figure 1). One possibility is to use ML in forecasting external input data for the ABM, which can then subsequently be used to inform agent behavior [FPV16, Pi16a, Pi16b, Sa16]. This is shown in the top half of figure 1, where ML is used to forecast aspects such as production, load and market price and provide these inputs to the agent. The second possibility is to use ML algorithms to implement the learning behavior of agents when they place bids on the market [KUP03, MGW18, Pe13]. This is shown in the bottom half of figure 1 where the learning behavior of agents could be either rule-based (the predefined strategies in figure 1) or through using RL algorithms.

It is also possible to use supervised learning techniques (as an alternative to RL) in a two-step approach to allow agents to place bids. Fischer [Fi18] describes this approach for financial markets where supervised learning is first used to build a predictive model using historical data, and then the forecasts from this predictive model are fed into a trading module to derive the trading action, e.g. buy or sell when the forecasted market price passes a certain threshold. There are a number of limitations in using supervised learning to directly place bids, which are discussed by Fisher [Fi18]. First, the optimization objective in the predictive model, i.e., the minimization of the forecast error, is not necessarily in line with the ultimate goal of the agent, e.g., the maximization of profits. Second, in most cases, only the forecast itself is used as an input, and additional valuable information that could be obtained from the feature space is discarded [Fi18, Mo98]. Finally, in the context of ABM with a large number of agents that interact dynamically, it is desirable to use lean algorithms that are computationally efficient. The use of RL as an alternative to supervised learning allows the forecast and the subsequent selection of a strategy to be carried out in one single step and both to be optimized in line with the objective of the agent [Fi18]. Therefore, RL and novel implementations of RL such as multi-agent RL and deep reinforcement learning (deep RL) are popular solutions to implement learning behavior in agents.

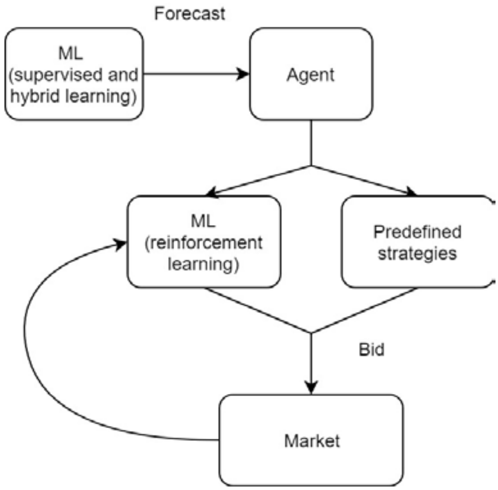

Figure 1: Possible use of ML and data-driven methods in ABM of energy markets

This paper aims to provide an overview of published literature on the common methods of integration of ML and data-driven methods in ABM of energy markets. While there are a number of published reviews on the use of ML for forecasting, e.g. [HF16] or on RL methods to improve decision making in agents, e.g. [WV08], there is no comprehensive overview on how different ML and data-driven methods can help improve ABM of LEM and the specific requirements for their integration.. In this paper, published journal articles (from year 2000 onwards) on integration of RL methods in ABM of energy markets are selected and presented in Section 1. In Section 2, we discuss the requirements of developing ML and data-driven methods for integration with ABM making reference to this selected literature. The conclusions are presented in the final section.

2 Brief summary of published literature on machine learning used in agent-based models of energy markets

Recent literature on ABM of energy markets includes the use of ML and other data based approaches to improve models and represent complexity in simulations. Selected journal articles on ABM of energy markets that include ML and data based methods are presented in table 1. For each reference, if ML is used in forecasting certain values, the subject of forecast, for example electricity market price, renewable generation, load, etc., is identified and noted in the second column of table 1. The type of ML algorithm used to derive or calculate the subject of forecast is noted in the third column of table 1. If RL is used, the kind of learning algorithm is identified and noted in the last column. The objective of RL algorithms in the selected references is to place bids on the market. Most references describe either the use of ML for forecasting, or RL for agents' bids on the market, however some references mention both cases. Even if forecasts are used as input data to the learning algorithm or to model agent behavior, the details of the forecasted data and the methods used for forecasting might not be specified in the article. In this case, the comment 'not specified' is noted in the relevant column.

2.1 Forecasting to inform decision-making

The bidding behavior of agents in the context of electricity or energy markets is often informed by the forecasted values of a number of inputs such as the forecasted load, generation and market price. Short-term forecasting methods are the most relevant when considering wholesale day-ahead or intra-day energy markets. Supervised learning methods and statistical methods are the most commonly used ML methods in forecasting [HF16]. In addition to forecasting based on historical weather, load and market price, the introduction of smart meters in many markets also provides a valuable source of more detailed data for forecasting loads [De11, Wa18]. In some cases unsupervised methods such as clustering are applied along with supervised learning algorithms or statistical methods to provide forecasts [Au18, FPV16, MW18].

A number of articles have reviewed ML methods for forecasting; however, these reviews do not consider the use of these forecasts for ABM or simulations. Table 1 includes some published articles where forecasting methods have been specifically developed for integration in ABM of energy markets. However, only few articles elaborate on the methods they use to derive the parameters that are used to inform the bidding behavior of agents. Since this is a new area of research, there is scope for further research on selection of ML algorithms for the particular case for forecasting as an input to ABM.

2.2 Multi-agent reinforcement learning for intelligent bidding

As discussed in the previous section, RL is a popular solution to implement learning behavior in agents. Although the agents can be endowed with behaviors designed in advance, they often need to learn new behaviors online such that the performance of the agent or of the whole multi-agent system gradually improves [Bu10, SW99, SV00]. In the case of electricity or energy markets, since the environment changes over time, a hardwired or pre-defined behavior of agents is inappropriate. The articles in table 1 apply a variety of RL algorithms with single or multiple agents to conduct experiments of different types of electricity market simulations. A broad spectrum of RL algorithms exist, e.g., model-free methods based on the online estimation of value functions, model-based methods (typically called dynamic programming), and model-learning methods that estimate a model, and then learn using model-based techniques [Bo10]. In the selected literature, mainly model free approaches have been applied, for example Q-learning and SARSA e.g. [Bo18, BEC18, EKS17, KUP03, PRD18, Pe13, Ya18] but some authors have also used model-based techniques, e.g. [BO01, VI08, Zh16].

In addition to the learning approach, another consideration is the definition of an appropriate formal goal for the learning. The articles in table 1 mainly focus on cases where agents act non-cooperatively to maximize their own interests. Mguni et al. find that while the lack of coordination produces stable outcomes or Nash equilibria, these are vastly suboptimal from a system perspective [Du08, Mg19]. Therefore, they propose an incentive-design method that modifies agents' rewards in a non-cooperative ABM that results in independent, self-interested agents choosing actions that produce optimal system outcomes in strategic settings.

Experience sharing, for instance agents exchanging information using communication, skilled agents serving as teachers for the learner, or the learner watching and imitating the skilled, can help agents with similar tasks learn faster and reach better performance [Bu10]. However, in the selected studies, there is no direct information exchange between agents, and the information flow is mainly directed from the market to the agents. Most of the studies consider the case of agents competing to maximize their own profits under different levels of market competition, however, Zhang et al. [Zh17] consider the case of optimal consensus control. Therefore, in the context of providing flexibility and encouraging consumption of electricity produced locally, it might be relevant to consider cases where the agents (prosumers and consumers) pursue cooperative strategies rather than purely competing strategies.

Reference	**Forecast: subject**	**Forecasting algorithm/ Derivation method**	**Learning algorithm**
Faia et al., 2016 [FPV16]	Electricity price in contracts	Hybrid (k-means and fuzzy logic)	Not specified
Aliabadi et al., 2017 [EKS17]	Locational marginal price at each node	DC- Optimal power flow problem	Q-learning
Pinto et al., 2016 [Pi16b]	Electricity market price	Support Vector Machines	Not specified
Kutschinski et al., 2003 [KUP03]	-	Not specified	Q-learning
Azadeh et al., 2010 [ASM10]	-	Not specified	Ant colony optimization
Zhang et al., 2017	-	Not specified	Adaptive dynamic

[Zh17]			programming
Mengelkamp at al., 2018 [MGW18]	Load	Standard profiles with error function	Erev-Roth reinforcement learning
Bunn & Oliveira, 2001 [BO01]	-	Not specified	Defined strategies
Visudhiphan & Ilic, 2008 [VI08]	-	Not specified	Defined strategies
Zhou et al., 2011[ZZW11]	Load, electricity price	Simulation model, polynomial cost function (producer)	Erev-Roth reinforcement learning
Yu et al., 2019 [Yu19]	-	Not specified	Experience-weighted attraction learning
Viehmann et al, 2018 [VLM18]	-	Not specified	Q-learning
Peters et al., 2013 [Pe13]	-	Not specified	State-Action-Reward-State-Action (SARSA)
Yang et al., 2018 [Ya18]	Load	k-means clustering	Q-learning
Patyn et al., 2018 [PRD18]	-	Not specified	Fitted Q-iteration with: a multilayer perceptron, a convolutional neural network and a long short-term memory neural network
Boukas et al., 2018 [BEC18]	-	Not specified	Q-learning, Q-function with a Neural Network
Boukas et al., 2018 [Bo18]	-	Not specified	Q-learning, Deep Q-Network
Chen et al., 2019 [CLS19]	Electricity market price	Extreme Machine Learning	Not specified

Tab. 1: A selection of published articles which use ML and data based methods in ABM of energy markets.

3 Requirements for the integration of machine learning and data-driven methods in multi-agent systems

3.1 Computational efficiency

Low computational demands mean lower costs, which increases the likelihood of automated bidding agents based on ML algorithms being deployed at prosumer's premises. In the selected literature, performance comparisons which include computational efficiency between different variations on algorithms which use the same RL approach are presented. For example, Patyn et al. [PRD18] use model-free RL to model the a heat pump agent which shifts loads in a day-ahead market to minimize daily electricity costs. They approximate the Q-function by three different neural architectures, a multilayer perceptron (MLP), a convolutional neural network (CNN) and a long short-term memory neural network (LSTM), and find that all architectures outperform a trivial thermostat controller and shift loads successfully after 20-25 days. In their modeled case, they do not find a significant difference in the performance of the MLP and the LSTM, both of which outperform the CNN model. However, they find that the MLP requires far less computation time. Pinto et al. [Pi16b] compare a support vector machines (SVM) based approach with artificial neural networks (ANN) to forecast the electricity market price. They show SVM methods provide similar results but take half the time of ANN. Finally, Mengelkamp et al. [MGW18] find the computational time for RL based strategies to be twice as high compared to bidding with random prices or with a selected fixed price. However, the computational time of their implementation of different variations of RL strategies differ by only 6%. Thus, they do not consider computational time as a criterion for selecting a particular strategy.

Deep RL or the use of deep neural networks within RL for value function approximations has also been shown to be successful in is in scaling up prior work in RL to high-dimensional problems. By means of representation learning, they can deal efficiently with the curse of dimensionality, unlike tabular and traditional non-parametric methods [Ar17, BCV13]. A relevant future research direction would be to compare the performance of dynamic programming approaches with deep RL approaches, since these are state of the art RL algorithms. The availability of open source implementations of different reinforcement algorithms (discussed in section 3.2) allows for the definition of standard benchmarks for testing new algorithms and evaluating new techniques in a standardized manner.

3.2 Learning curve or difficulty of implementation of machine learning methods

The learning curve in implementing ML methods is an important consideration because ABM developers cannot focus exclusively on the implementation of these methods but also need to consider other aspects of modeling such as interactions between agents and the mechanics of market clearing. Therefore, the availability of standard libraries, examples and detailed documentation are a consideration when selecting the method for implementation. While most publications do not mention the details and the use of standard libraries used in their implementation of ML algorithms, a wide selection of open source libraries are available in common programming languages to implement supervised, unsupervised, and RL algorithms.

Some common libraries in Python to implement RL are OpenAI Gym or Universe, RLLib, Coach, TensorForce, Keras-RL, PyBrain, RLPy [Ge13, In19, Op19, Pl16, Ra19, Re19, Sc10]. Libraries implemented in Java for RL are BURLAP, RL4J, RL-Glue [Ch19a, Sk19, TW09] and packages for R are ReinforcementLearning and MDPtoolbox [CH19b, PF19].

MATLAB also offers a number of libraries to implement ML algorithms, Pinto et al. [Pi16b] use it to develop their SVM approach to forecast market prices and Mengelkamp et al. [MGW18] use it to implement RL algorithms. Chen & Su [CS18] implement their RL algorithm in Python, and Lamperti et al. [La18] also use Python to implement their model calibration approach.

In addition to standardized frameworks for implementation, another consideration with respect to the difficulty of implementation is the definition of an appropriate formal goal for the learning multi-agent system. As discussed in Section 2.2, a common approach is to apply single-agent Q-learning to the multi-agent case where the learned Q-functions only depend on the current agent's action without being aware of the other agents. Busoniu et al. [Bu10], find that one important research direction is understanding the conditions under which single-agent RL works in mixed stochastic games, especially given the preference towards using single-agent techniques for multi-agent systems in practice.

3.3 Flexibility or adaptability of the machine learning algorithms in a multi-agent system

ML models can have different learning rates with different datasets, and need to be tuned so that they can optimally solve the ML problem. The measures used to tune a model are called hyperparameters. In the context of ML providing input data to an agent in an ABM, it is important that the hyperparameters can be easily set and adjusted to allow selection between accuracy and computational time, for example. In the Multi-Agent System for Competitive Electricity Markets (MASCEM) platform developed by Santos et al., the management of the system to adapt its execution time to the purpose of the simulation is performed by means of a fuzzy process [Sa16]. Standard libraries in Python, for example scikit-learn, allow hyperparameter optimization using several methods like grid search, random search and Bayesian optimization. These methods could be integrated in the architecture of the ABM platform to enable adaptability of the ML algorithms.

In the context of RL algorithms, the algorithms can be tuned by choosing the learning rate, selecting the resolution of the value function, choosing how often to update the representation of the value function, and making tradeoffs between exploring to improve the learning model and exploring to improve the learning policy [AS02]. The consequences of these choices are greatly influenced by which RL approach is selected and the specific details of how the algorithm is implemented. Atkeson and Santamaria [AS02] find that there are fewer parameter choices to make in model-based RL. The (hyper) parameter values in RL also influence whether convergence is achieved and how quickly it is achieved.

3.4 Robustness

The agent's perception of the environment may vary, and therefore the robustness of an ML algorithm is an important consideration. Multi-agent RL is inherently robust because if one or more agents fail in a multi-agent system, the remaining agents can take over some of their tasks [Bu10]. Other properties of multi-agent RL are stability and adaptation: an opponent-independent algorithm converges to a strategy that is part of an equilibrium solution regardless of what the other agents are doing while an opponent-aware algorithm learns models of the other agents and reacts to them using some form of best response. Algorithms focused on stability (convergence) only are typically unaware and independent of the other learning agents [Bu10]. Common methods to measure robustness are convergence time and change in output/convergence values across multiple runs. In Rosen & Madlener [RM13], tests which consider the speed of convergence are used to quantify robustness of the algorithm. In Viehmann et al. [VLM18] each model is run multiple times with varying seeds to check for multiple stable outcomes and robustness of results. Peters et al. [Pe13] consider noise injection, to alleviate overfitting and improve generalization in supervised settings.

None of the selected articles compares the robustness of different algorithms. However, a general understanding is that ABM with agents that are unaware or do not directly interact with the other agents converge more easily, while in other cases reward functions or other criteria need to be specifically defined in order to achieve convergence. For example, Zhou et al. [ZWL18] use step length control and learning process involvement to facilitate convergence and also define a last-defense mechanism (ending the simulation after a pre-defined finite number of iterations, regardless if convergence is achieved or not) to handle divergence.

4 Conclusions

In this paper, we provide an overview of published literature on the common methods of integration of ML and data-driven methods in ABM of energy markets. We discuss some important requirements for this integration and present the methods used in published articles to address these requirements.

Since the integration of ML methods in ABM is a relatively new area of research, there are few articles which discuss the methods of such integration and the benefits it can offer. The purpose of our contribution is to provide a first (to the best of our knowledge) review of such novel approaches which may serve as a starting point for future research efforts.

Further case studies are required for a clear comparison considering the highlighted dimensions as well as additional dimensions. As discussed in section 2.1, further research on the selection of suitable and efficient ML algorithms specifically to provide inputs for ABM and simulations is necessary. In addition, a formalized architecture and a common module which can use data inputs, e.g. weather related and load related parameters such as temperature-humidity index, wind chill index, etc. used for the forecasting algorithms would improve the efficiency and modularity of integrating ML with the ABM.

With respect to implementing learning behavior in agents, a number of future research areas have been identified: comparison of the robustness of different algorithms, the suitability and selection of RL algorithms specific to the use case of bidding on markets, and identification of algorithms which can better represent cooperative strategies, and conversely, non-cooperative agent strategies. Finally, it is also relevant to conduct experiments where the agents (prosumers and consumers) pursue cooperative strategies rather than purely competing strategies. In the reviewed literature, it is difficult to compare the efficiency of different RL algorithms because they have been implemented in different types of ABM, with different assumptions and market dynamics. Experiments that compare different RL approaches but with the same market assumptions would be valuable, as they would help in benchmarking the different algorithms and provide the possibility to identify which algorithms are more suited for or efficient in specific market designs.

In summary, ML can be used in ABM, for example to forecast input parameters which agents can use in their decision making, and, for the learning as the simulation goes along (i.e. for RL). There is scope for further research and definition of standardized test cases on all types of coupling, as well as definition of standardized methods to evaluate new techniques. Numerous open source libraries and frameworks allow such implementation to be feasible and efficient.

Bibliography

[Al10] Alpaydin, E.: Introduction to Machine Learning. 2nd. ed.: The MIT Press, 2010.

[Ar17] Arulkumaran, K. et al.: Deep reinforcement learning: A brief survey. In: IEEE Signal Processing Magazine vol. 34, Nr. 6, pp. 26–38, 2017.

[AS02] Atkeson, C.G. ; Santamaria, J.C.: A comparison of direct and model-based reinforcement learning, Nr. April, pp. 3557–3564, 2002.

[Au18] Auder, B. et al.: Scalable Clustering of Individual Electrical Curves for Profiling and Bottom-Up Forecasting. In: Energies vol. 11, Nr. 7, p. 1893, 2018.

[ASM10] Azadeh, A.; Skandari, M.; Maleki-Shoja, B.: An integrated ant colony optimization approach to compare strategies of clearing market in electricity markets: Agent-based simulation. In: Energy Policy vol. 38, Elsevier, Nr. 10, pp. 6307–6319, 2010.

[BCV13] Bengio, Y.; Courville, A.; Vincent, P.: Representation learning: A review and new perspectives. In: IEEE Transactions on Pattern Analysis and Machine Intelligence vol. 35, Nr. 8, pp. 1798–1828, 2013.

[Bo18] Boukas, I. et al.: Intra-day Bidding Strategies for Storage Devices Using Deep Reinforcement Learning. In: International Conference on the European Energy MarketISBN 9781538614884, 2018.

[BEC18] Boukas, I.; Ernst, D.; Cornelusse, B.: Real-Time Bidding Strategies from Micro-Grids Using Reinforcement Learning. In: CIRED Workshop 2018, Nr. 0440, pp. 7–8, 2018.

[BOR17] Bremdal, B.A.; Olivella, P.; Rajasekharan, J.: EMPOWER: A network market approach for local energy trade. In: 2017 IEEE Manchester PowerTech, 2017, pp. 1–6

[BO01] Bunn, D. ; Oliveira, F.S.: Agent-Based Simulation—An Application to the New Electricity Trading Arrangements of England and Wales. In: Ieee Transactions on Evolutionary Computation, vol. 5, Nr. 5, pp. 493–503, 2001.

[Bu10] Busoniu, L. et al.: Multi-agent reinforcement learning: An overview. In: (Srinivasan, D.; Jain, L. C. eds.): Innovations in Multi-Agent Systems and Applications -- 1, Studies in Computational Intelligence. vol. 310. Berlin, Germany : Springer, 2010, pp. 183–221

[Ch19a] Chades, I. et al.: Brown-UMBC Reinforcement Learning and Planning (BURLAP). http://burlap.cs.brown.edu/faq.html#cite. - accessed 22/4/2019

[Ch19b] Chades, I. et al.: MDPtoolbox: Markov Decision Processes Toolbox. https://cran.r-project.org/package=MDPtoolbox, accessed: 22/4/2019

[CLS19] Chen, K.; Lin, J.; Song, Y.: Trading strategy optimization for a prosumer in continuous double auction-based peer-to-peer market: A prediction-integration model. In: Applied Energy vol. 242, pp. 1121–1133, 2019.

[CS18] Chen, T. ; Su, W.: Indirect Customer-to-Customer Energy Trading with Reinforcement Learning. In: IEEE Transactions on Smart Grid vol. PP, IEEE, Nr. c, p. 1, 2018.

[De11] De Silva, D. et al.: Semi-supervised classification of characterized patterns for demand forecasting using smart electricity meters. In: 2011 International Conference on

Electrical Machines and Systems, ICEMS 2011, IEEE, pp. 1–6ISBN 9781457710445, 2011.

[Du08] Dubey, P.: Inefficiency of Nash Equilibria. In: Mathematics of Operations Research vol. 11, Nr. 1, pp. 1–8, 2008.

[EKS17] Esmaeili Aliabadi, D.; Kaya, M.; Sahin, G.: Competition, risk and learning in electricity markets: An agent-based simulation study. In: Applied Energy vol. 195, Elsevier Ltd, pp. 1000–1011, 2017.

[Fa14] Faber, I. et al.: Micro-energy markets: The role of a consumer preference pricing strategy on microgrid energy investment. In: Energy vol. 74, Elsevier Ltd, Nr. C, pp. 567–575, 2014.

[FPV16] Faia, R.; Pinto, Z.; Vale, Z.: Dynamic Fuzzy Clustering Method for Decision Support in Electricity Markets Negotiation. In: ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal vol. 5, Nr. 1, p. 23, 2016.

[Fi18] Fischer, T.G.: Reinforcement learning in financial markets-a survey, 2018

[Ge13] Geramifard, A. et al.: RLPy: The Reinforcement Learning Library for Education and Research, 2013.

[HF16] Hong, T. ; Fan, S.: Probabilistic electric load forecasting: A tutorial review. In: International Journal of Forecasting vol. 32, Elsevier B.V., Nr. 3, pp. 914–938, 2016.

[In19] Intel AI Lab: Reinforcement Learning Coach. https://nervanasystems.github.io/coach/., accessed: 22/4/2019

[KUP03] Kutschinski, E.; Uthmann, T.; Polani, D.: Learning competitive pricing strategies by multi-agent reinforcement learning. In: Journal of Economic Dynamics and Control vol. 27, Nr. 11–12, pp. 2207–2218, 2003.

[La18] Lamperti, F. et al.: Agent-based model calibration using machine learning surrogates. In: Journal of Economic Dynamics and Control vol. 90, pp. 366–389, 2018.

[Me18] Mengelkamp, E. et al.: Designing microgrid energy markets: A case study: The Brooklyn Microgrid. In: Applied Energy vol. 210, pp. 870–880, 2018.

[MGW18] Mengelkamp, E.; Gärttner, C.; Weinhardt, C.: Intelligent Agent Strategies for Residential Customers in Local Electricity Markets, pp. 97–107ISBN 9781450357678, 2018.

[MW18] Mengelkamp, E.; Weinhardt, C.: Clustering Household Preferences in Local Electricity Markets, pp. 538–543ISBN 9781450357678, 2018.

[Mg19] Mguni, D. et al.: Coordinating the Crowd: Inducing Desirable Equilibria in Non-Cooperative Systems, Nr. Id, 2019.

[Mo98] Moody, J. et al.: Performance functions and reinforcement learning for trading systems and portfolios. In: Journal of Forecasting vol. 17, Nr. December 1997, pp. 441–470, 1998.

[Mo18] Morstyn, T. et al.: Using peer-to-peer energy-trading platforms to incentivize prosumers to form federated power plants. In: Nature Energy vol. 3, Springer US, Nr. 2, pp. 94–101, 2018.

[Op19] OpenAI: *Gym*. https://gym.openai.com/., accessed: 22/4/2019

[PRD18] Patyn, C.; Ruelens, F.; Deconinck, G.: Comparing neural architectures for demand response through model-free reinforcement learning for heat pump control. In: 2018 IEEE International Energy Conference, ENERGYCON 2018, IEEE, pp. 1–6ISBN 9781538636695, 2018.

[Pe13] Peters, M. et al.: A reinforcement learning approach to autonomous decision-making in smart electricity markets. In: Machine Learning vol. 92, Nr. 1, pp. 5–39, 2013.

[Pi16a] Pinto, T. et al.: Adaptive Portfolio Optimization for Multiple Electricity Markets Participation. In: IEEE Transactions on Neural Networks and Learning Systems vol. 27, Nr. 8, pp. 1720–1733, 2016.

[Pi16b] Pinto, T. et al.: Support Vector Machines for decision support in electricity markets' strategic bidding. In: Neurocomputing vol. 172, Elsevier, pp. 438–445, 2016.

[Pl16] Plappert, M.: Keras-RL. https://keras-rl.readthedocs.io/en/latest/., accessed: 22/4/2019

[PF19] Pröllochs, N. ; Feuerriegel, S.: Reinforcement Learning in R. https://cran.r-project.org/web/packages/ReinforcementLearning/vignettes/ReinforcementLearning.html. - accessed 22/4/2019

[Ra19] The Ray Team Revision: RLlib: Scalable Reinforcement Learning. https://ray.readthedocs.io/en/latest/rllib.html., accessed: 22/4/2019

[Re19] reinforce.io: TensorForce - modular deep reinforcement learning in TensorFlow. https://tensorforce.readthedocs.io/en/latest/., accessed: 22/4/2019

[RKF16] Ringler, P. ; Keles, D. ; Fichtner, W.: Agent-based modelling and simulation of smart electricity grids and markets - A literature review. In: Renewable and Sustainable Energy Reviews vol. 57, Elsevier, pp. 205–215ISBN 1364-0321, 2016.

[RM13] Rosen, C. ; Madlener, R.: An auction design for local reserve energy markets. In: Decision Support Systems vol. 56, Elsevier B.V., Nr. 1, pp. 168–179, 2013.

[Sa16] Santos, G. et al.: MASCEM: Optimizing the performance of a multi-agent system. In: Energy vol. 111, Elsevier Ltd, pp. 513–524, 2016.

[Sc10] Schaul, T. et al.: PyBrain. In: Journal of Machine Learning Research, 2010.

[SW99] Sen, S. ; Weiss, G.: Learning in Multiagent Systems. In: Weiss, G. (ed.): Multiagent systems: A modern approach, Cambridge, MA, USA : MIT Press, 1999 — ISBN 0-262-23203-0, pp. 259–298

[Se07] Sensuß, F. et al.: Agent-based Simulation of Electricity Markets - A Literature Review. In: Energy Studies Review vol. 15, Nr. 2, p. 44, 2007.

[Sk19] Skymind: Deep Learning for Java. https://deeplearning4j.org/., accessed: 22/4/2019

[So18] Sousa, T. et al.: Peer-to-peer and community-based markets: A comprehensive review, 2018.

[SV00] Stone, P. ; Veloso, M.: Multiagent systems: a survey from a machine learning perspective. In: Autonomous Robots vol. 8, Nr. 3, pp. 345–383ISBN 0929-5593, 2000.

[TW09] Tanner, B. ; White, A.: RL-Glue: Language-Independent Software for Reinforcement-

Learning Experiments. In: J. Mach. Learn. Res. vol. 10, pp. 2133–2136, 2009.

[VLM18] Viehmann, J.; Lorenczik, S.; Malischek, R.: Multi-unit Multiple Bid Auctions in Balancing Markets : an Agent-based Q-learning Approach, Nr. 18, 2018.

[VI08] Visudhiphan, P. ; Ilic, M.D.: Dynamic games-based modeling of electricity markets, pp. 274–281 vol.1ISBN 0780344030, 2008.

[Wa18] Wang, Y. et al.: Review of Smart Meter Data Analytics: Applications, Methodologies, and Challenges. In: IEEE Transactions on Smart Grid vol. 3053, Nr. June 2017, pp. 1–24, 2018.

[WV08] Weidlich, A. ; Veit, D.: A critical survey of agent-based wholesale electricity market models. In: Energy Economics vol. 30, Nr. 4, pp. 1728–1759ISBN 0140-9883, 2008.

[Ya18] Yang, Y. et al.: Recurrent Deep Multiagent Q-Learning for Autonomous Agents in Future Smart Grid Extended Abstract. In: Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), pp. 2136–2138, 2018.

[Yu19] Yu, Q. et al.: The Strategy Evolution in Double Auction Based on the Experience-Weighted Attraction Learning Model. In: IEEE Access vol. 7, pp. 16730–16738, 2019.

[Zh16] Zhang, C. et al.: A Bidding System for Peer-to-Peer Energy Trading in a Grid-connected Microgrid. In: Energy Procedia vol. 103, Elsevier B.V., Nr. April, pp. 147–152, 2016.

[Zh17] Zhang, H. et al.: Data-Driven Optimal Consensus Control for Discrete-Time Multi-Agent Systems with Unknown Dynamics Using Reinforcement Learning Method. In: IEEE Transactions on Industrial Electronics vol. 64, IEEE, Nr. 5, pp. 4091–4100, 2017.

[Zh18] Zhang, C. et al.: Peer-to-Peer energy trading in a Microgrid. In: Applied Energy vol. 220, Elsevier, Nr. December 2017, pp. 1–12, 2018.

[ZWL18] Zhou, Y.; Wu, J.; Long, C.: Evaluation of peer-to-peer energy sharing mechanisms based on a multiagent simulation framework. In: Applied Energy vol. 222, Elsevier, Nr. February, pp. 993–1022ISBN 0046405003, 2018.

[ZZW11] Zhou, Z.; Zhao, F.; Wang, J.: Agent-based electricity market simulation with demand response from commercial buildings. In: IEEE Transactions on Smart Grid vol. 2, IEEE, Nr. 4, pp. 580–588, 2011.