Machine learning for agile and self-adaptive congestion management in active distribution networks

Citation for published version (APA):

Babar, M., Roos, M., & Nguyen, P. (2019). Machine learning for agile and self-adaptive congestion management in active distribution networks. In *2019 IEEE International Conference on Environment and Electrical Engineering and 2019 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe)* [8783624] Institute of Electrical and Electronics Engineers. <https://doi.org/10.1109/EEEIC.2019.8783624>

DOI:

[10.1109/EEEIC.2019.8783624](https://doi.org/10.1109/EEEIC.2019.8783624)

Document status and date:

Published: 01/08/2019

Document Version:

Accepted manuscript including changes made at the peer-review stage

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Machine Learning for Agile and Self-Adaptive Congestion Management in Active Distribution Networks

M. Babar

*Electrical Energy Systems
Eindhoven University of Technology
Eindhoven, the Netherlands
m.babar@tue.nl*

M.H. Roos

*Asset Management
Enexis Netbeheer
's-Hertogenbosch, the Netherlands*

P.H. Nguyen

*Electrical Energy Systems
Eindhoven University of Technology
Eindhoven, the Netherlands*

Abstract—Although congestion management via Demand Response (DR) has gained sufficient popularity recently, there are still some fundamental impediments to achieve a trade-off between demand flexibility scheduling and demand flexibility dispatch for congestion management. To find a solution to the challenge, the paper introduces the concept and design of an Agile Net, which is an agile control strategy for congestion management. The model of Agile Net has three cores. First, it perceives the network environment by using the concept of demand elasticity. Second, it possesses an online model-free learning technique for the management of network externality, such as congestion. Third, it enables distributed system scalability. The efficiency of the proposed Agile Net is investigated by extending the simulation tool for DR paradigm for a generic low-voltage network of the Netherlands. Simulation results reveal a significant reduction in congestion over a year while confirming expected levels of performance.

Index Terms—Agile demand response, Model-Free, Multi-agent system, Reinforcement learning, Transactive energy.

I. INTRODUCTION

Congestion in distribution networks, which might occur due to the high penetration of distributed energy resources (DERs) and an increase in domotics, is a big challenge to the distribution system operator (DSO). Numerous demand response (DR) strategies have already been investigated to mitigate congestion and improve operations at low-voltage (LV) level. The investigated DR strategies for congestion management (CM) can be classified into (1) implicit DR using dynamic pricing [1]–[9], and (2) explicit DR using transactive energy [10], [11].

In implicit DR, price tariffs or mutual contracts are used to manage congestion, refers to avoid the thermal overload of the system by reducing peak loads, by demand flexibility scheduling. In the recent literature, CM via implicit DR consists of the dynamic tariff [1]–[4], distribution capacity market [5], [6], intra-day shadow price [7], and flexible service market [8]. However, most of the solutions modeled DR only for demand flexibility scheduling, thus DR can be predicted with a negligible error margin. Different types of locational marginal

price or distribution congestion price have been investigated to address the problem of thermal constraint violations in distribution networks [12]. However, implicit DR methods mainly depend on the willingness and availability of demand flexibility and are therefore may not always be available to resolve the congestions completely.

Explicit DR provides a paradigm for DSO to achieve similar effects by combining direct and dynamic pricing methods with more efficient and robust control of the network in real-time [10], [11]. In [11], the dynamic thermal model of the transformer has been used to quantify the cost incurred for overloading. Transactive energy is used to coordinate the entities and procure flexibility to alleviate the overloading cost. A direct control has been adopted to limit connection capacity when congestion cannot be resolved through the market. A local market framework incorporating day-ahead and real-time operations is proposed in [9] that aims to minimize the cost of network support through the voluntary and compulsory participation of the prosumers. A spatially distributed hierarchical control architecture is discussed in [13], that incorporates the network reconfiguration possibilities along with market-based flexibility. The proposed DR paradigm determines optimal dispatch of a grid-connected DERs and dispatchable home appliances (referred to as domotics) providing flexibility services to the market and the network. However, the recently available explicit DR strategies have limitations. More specifically, the DSO's controller/agent takes actions shortly after the congestion has perturbed the network. As a result, there is usually a delay in control between the transformer's state and operator's action to mitigate the congestion. So, it is inferred that if the demand is highly volatile due to DERs and domotics, DSO may not be able to achieve the desired state. Thus, DSO requires acting proactively to cope this challenge.

Reinforcement learning (RL) approaches in DR have attracted significant attention as a possible solution to address the limitations associated with the scheduling [14]–[16], forecasting [17], and dispatch [13], [18] of flexible demand. Reinforcement

learning (RL) is concerned with how an agent ought to take actions in a dynamic environment to maximize long-term reward [19].

In this research, an agile solution for real-time congestion management is introduced by incorporating an advanced agile DR with a transaction energy based on demand flexibility constructs while considering grid reliability constraints. The solution is named as agile because it possesses attributes that are already well-known to computer society for many years in the area of process development. Moreover, the term was first coined in power system by Paul A. Centolella at Analysis Group, an economic and strategy consulting firm, in 2012 [20]. Therefore, in this paper, the agile approach to network congestion is referred to as Agile Net because it complements the advantages of the real-time explicit DR and continual reinforcement learning capability, and ensures a more reliable operation of the network. The Agile Net is different from other approaches because it makes use of the distributed intelligence within an agent-based environment having a peer-to-peer networking. Moreover, the price-elasticity of demand is conceptualized to estimate the impact of congestion on the network. A detail mathematical modeling of the Agile Net is formulated as a milestone in the development of online model-free learning techniques for network operations using demand flexibility. A detailed case study is performed on a modified Dutch LV test network to investigate the efficiency of the proposed approach. The main contributions of the paper are as follows:

- an agile approach, which complements transactive energy and real-time congestion management, is defined;
- a price-elasticity of demand is proposed to incorporate the effects of DERs and domotics in the LV networks;
- online model-free machine learning is enabled via agent-based distributed architecture; and
- proactive decision-making is applied to identify proper incentivization.

The paper is organized as follows: Section II presents an overview of the Agile Net along with the mathematical formulation. Section III provides the description of the test network and the associated numerical assumptions. Furthermore, it also discusses the findings of the numerical simulations. Finally, section IV concludes the results with some future recommendations.

II. THE AGILE NET

The Agile Net consists of a DSO, Host, Percept and Learning agents which communicate with each other and with the aggregator via peer to peer communication as shown in Fig. 1. The DSO agent is an arbitrator who is connecting other agents of Agile Net to the aggregator. Potentially, it does not take part in the decision-making process, though it passes on the final of Agile Net to the aggregator. The Host agent is taking charge of real-time network monitoring and has a direct impact on management decisions. A real-time state should provide insights into the demand consumption of the users connected to the host network and their participation in a

DR service provided by the aggregator. The Percept agent is responsible for defining reward and report expected next state of the host agent to the learning agent. The Learning agent is a self-learning agent responsible for CM based on RL. It fits perfectly within Agile Net's feedback driven, interactive framework. RL offers opportunities for highly autonomous and adaptive CM. It assumes no prior knowledge about the aggregator's running environment.

Due to the extensible, scalable and model-free learning design, the agile net is very suitable for real-time congestion management and easy to use.

In the remainder of this section, the mathematical design of the DSO, Host, Percept and Learning agents are described.

Fig. 1. The architecture of Agile Net, illustrating peer to peer interactions among agents and an aggregator within Agile DR architecture. (a) aggregated bid(s). (b) aggregated bid(s) from aggregator. (c) incentive signal. (d) transformer's status (power, etc.). (e) the incentive signal(s) sent by DSO agent to aggregator. (f) request for network performance parameters. (g) send network performance parameters (h) estimated next state and reward. (i) actions for learning. (j) update policy for decision making. (k) update the probabilities associated with network performance parameters.

A. DSO agent

The management operation by DSO agent to an aggregator is defined as a combination of three meta operations, i.e., no-operation (nop), decrease or increase in the price signal via incentivization.

B. Host agent

The real-time network state is defined as a quad-tuple vector.

$$s^k \doteq \{k, \phi^k, \varphi^k, \mathcal{P}^k\}, \forall k \in \mathcal{K}, \forall s^k \in \mathcal{S} \quad (1)$$

where $k, \phi^k, \varphi^k, \mathcal{P}^k$ denote the time interval of operation, normalized value of congestion, the probability of congestion occurrence, and the probability density function of normalized price signal during the given time interval $k : k \in \mathcal{K}$, respectively. Assuming there is no loss of information, k is referred to as a specific interaction granularity of 15 minutes for a time horizon K . In this way, the host agent maintains state space of the network. At each interval, states k and ϕ^k are monitored and updated in the state space. However, state φ^k is updated concerning ϕ^k as follows:

$$\varphi^k = \begin{cases} (1 + \eta)\varphi^k & , |\phi^k| > \phi^{nom} \\ (1 - \eta)\varphi^k & , \text{otherwise} \end{cases} \quad (2)$$

where $\eta = 1/TR_{kVA}$ is a deferral factor that represents maximal flexible demand during an interval, while TR_{kVA} is maximum transformer rating. Moreover, ϕ^{nom} defines the nominal operational boundary of a transformer in a network. Depending on the network utilization, the DSO decides the most suitable value of ϕ^{nom} : $\phi^{nom} \in (0, 1)$. For instance, if the network has 400kVA transformer and the DSO decides 250kVA as the nominal operational boundary of the transformer, then $\phi^{nom} = \frac{250kVA}{400kVA} = 0.625$. Thus, as discussed in [21], the network utilization can be simply categorized into three operational regions; namely, nominal (when $|\phi^k| \leq \phi^{nom}$), alert (when $|\phi^k| \leq 1$) and emergency (when $|\phi^k| > 1$).

Host agent finds the normalized price signal (Γ^{k*}) and updates respective probability distribution function ($\mathcal{P}^k(\Gamma_j^k)$) that will be used by the Percept agent to find expected price signal during the learning process. Assume there is J number of price signals, such that $\Gamma_{\mathcal{J}}^k = \{\Gamma_1^k, \dots, \Gamma_j^k, \dots, \Gamma_J^k\}$. Each price signal Γ_j^k is updated with a probability $\mathcal{P}^k(\Gamma_{\mathcal{J}}^k)$. Initially, all probabilities are equal, however, with the passage of time the host agent updates $\mathcal{P}^k(\Gamma_j^k)$ as follows:

$$\mathcal{P}^k(\Gamma_j^k) = \begin{cases} \mathcal{P}^k(\Gamma_j^k) + \eta(1 + \mathcal{P}^k(\Gamma_j^k)) & , \Gamma_j^k = \Gamma^{k*} \\ \mathcal{P}^k(\Gamma_j^k) - \eta\mathcal{P}^k(\Gamma_j^k) & , \text{otherwise} \end{cases} \quad (3)$$

C. Percept agent

1) *Reward Function*: The reward ought to explicitly punish demand scheduling by the aggregator that leads to congestion, meanwhile encouraging actions that mitigate or cause no congestion. Whenever there is a conflict in the aggregated demand, e.g., the scheduled power becomes more than the transformer capacity in the response of an action, the percept agent sets reward to -1 (penalty) for the learning agent. Otherwise, it sets reward to 0 (neutral). Mathematically,

$$r = \begin{cases} -\varphi^k \times \left| \frac{\phi^k - \phi^{nom}}{\phi^{nom}} \right| - 1 & , |\phi^k| > \phi^{nom} \\ 0 & , \text{otherwise} \end{cases} \quad (4)$$

2) *A responsive environment*: The price elasticity of demand is considered to create a responsive environment that directly reflects network performance and DR to the learning agent in the state transition. The elasticity of demand is defined as the demand sensitivity concerning price, such as:

$$\varepsilon_{kk'} = \frac{\Gamma^{k'}}{l^k} \frac{\partial l^k}{\partial \Gamma^{k'}} \quad (5)$$

where ε is the elasticity coefficient indicates DR penetration, l^k is flexible demand during time interval k , and Γ^k is the normalized price indicator/signal of the spot price or any other dynamic pricing yielded by the aggregator. As discussed in [22], the elasticity of demand of the current k^{th} interval versus an interval $k' : k' \neq k, \forall k' \in \mathcal{K}$. From an economic point of view, ε_{kk} represents the self-elasticity and $\varepsilon_{kk'}$ corresponds to the cross-elasticity. In numerous economic textbooks, price elasticity of demand can be generalized as a downward slope of a demand curve at a given time. With these considerations,

the responsive virtual environment codifies the elasticity of demand into (6) for the enhanced iterative process. The detailed derivation is explained in [22], [23].

$$\varepsilon_{kk'} = \frac{-2m^{k^2}\Gamma^{k'} + m^k c^k}{\sqrt{c^{k^2} + 4m^k[-m^k\ddot{\Gamma} + c^k\dot{\Gamma} - C_E^k]}} \times \frac{\Gamma^{k'}}{-m^k\Gamma^k + c^k} \quad (6)$$

3) *State transition*: The environment provided by the percept agent helps the learning agent to perform iterative actions for the learning purposes. In response to an action, the environment provides next possible state of congestion (ϕ_{n+1}^k , as shown in (7)) as well as a reward (r) to the learning agent.

$$\phi_{n+1}^k = \varphi^k \eta b^k \left\{ 1 + \varepsilon_{kk} \frac{(1 + \phi_n^k) i_n^k}{\Gamma_n^k} + \sum_{k': k' \neq k}^K \varepsilon_{kk'} \frac{(1 + \phi_n^{k'}) i_n^{k'}}{\Gamma_n^{k'}} \right\} \quad (7)$$

D. Learning agent

A RL problem is usually modeled as a Markov Decision Process (MDP). As discussed, for a state space $\mathcal{S}$ and a set of actions $\mathcal{I}_{\mathcal{S}}$ for a given interval k , MDP is defined by the transition function $f : \mathcal{S} \times \mathcal{I}_{\mathcal{S}} \times \varphi \times \mathcal{P}(\Gamma) \rightarrow \mathcal{S}$ and an immediate reward function $r : \mathcal{S} \times \mathcal{I}_{\mathcal{S}} \times \mathcal{S} \rightarrow \mathbb{R}$. ($\varphi \times \mathcal{P}(\Gamma)$) is the set of possible realizations of a random process. The general evolution of the system is equivalently written as $s_{n+1} \sim \mathcal{PS}(\cdot | s_n, i_n)$, which highlights that the next state of the system follows a probability distribution that is conditional on the current state and on the action taken at the corresponding time step. At each step n , the learning agent perceives its current state $s_n \in \mathcal{S}$ and the available action set $\mathcal{I}_{\mathcal{S}}$. By taking action $i_n \in \mathcal{I}_{\mathcal{S}}$, the agent transits to the next state s_{n+1} and receives an immediate reward r from the environment. The optimal policy is simple, i.e., it always select the action i_n that maximizes the value function $Q(s_n, i_n)$ at state s_n . Finding the optimal policy is equivalent to obtain an estimation of value function which approximates its actual value. The estimation of $Q(s_n, i_n)$ can be updated during each iteration as follows:

$$Q(s_n, i_n) = Q(s_n, i_n) + \alpha[r + Q(s_{n+1}, i_{n+1}) - Q(s_n, i_n)] \quad (8)$$

where α is the learning rate. Herein, the ϵ -greedy policy is used to design the RL agent. With a small probability ϵ , the agent picks up a random action and follows the best policy it has found for the rest of the time.

In RL, actions in continuous space remains an open research problem; this paper also limits the learning agent to discrete actions. The price signal Γ is incremented or decremented by one at a time. Although the actions are limited to discrete incentives, the state itself can render the Q value function intractable. Due to its critical impact on the learning performance, there are many studies on the Q function approximation [19], [24]–[26]. However, after carefully reviewing these works, the tile coding technique, discussed in [27], is considered for the estimate of the Q function because it outperforms online learners. Moreover,

tile coding technique is a state discretization method which serves as a state-action pair function approximator for several different learning and control tasks.

Fig. 2. Tile Coding based Q table

III. EXPERIMENT DESIGN AND SIMULATION RESULTS

A. Configuration

A Dutch LV network is considered for simulation purposes. This network, is a holiday park at Bronsbergen, consists of residential loads and PV systems, each connected to a 400kVA 0.4/10kV transformer. The network has 210 individual households, 108 out of which have roof-mounted solar panels that can generate 315kW_p. The houses are supplied via four feeders with the length of 800m, 900m, 350m, and 450m, and the R/X ratio of the cables used in the LV network varies from 1 up to 8. The LV network of Bronsbergen is developed in MATLAB to perform load simulations along with power flow solution. The network has been used in this work for studying the effects of incentivization under different cases, given that every household of the network is committed to an aggregator.

B. Methodology

An attempt has been made to create dynamic variations in demand to evaluate the efficacy of the algorithm. The base load profiles are modeled by using a Markov chain Monte Carlo approach based on time-of-use surveys, statistics on appliance characteristics, and weather variables [28]. On the other hand, each appliance is modeled separately as an agent, which assesses the amount of flexible load available on a 15 min interval, and then shares its bid (or value proposition) to an upstream agent. The details and specifications of appliance agents are discussed in [13]. Furthermore, the given network is augmented such that up to 60% of the households are connected to PV systems, 80% of the households have electric vehicles

(EV), and every household has at least a controllable refrigerator and a washing machine. Moreover, the temporal constraints of the EV's and the washing/drying loads are assumed to be flexible, such that washing load can be shifted for up to 8 hours, while the EV load should be fully charged when the EV leaves.

To understand the performance of RL-based CM, two variations of Agile Nets are simulated. One with an initialization of the management policy and the other without any initialization that is represented as RL w/ init and RL w/o init, respectively. The results are compared to a base case without CM and a case without Learning agent. The initial policy was obtained by simulating for 10 days, during which the learning agent interacted with the environment with only exploration actions. Moreover, the granularity of time interval for simulations is set to 15 min (that implies 96 intervals per day).

C. Congestion Management performance

This subsection discusses the effectiveness of Agile Net in managing congestion by exploiting demand flexibility through a single aggregator.

1) *Learning time*: Fig. 3 plots the performance of Agile Net and its learning variations in 10 days simulation in the form of cumulative distribution over average learning duration. Cumulative distribution curves provide a measure of how much a learning time accumulates, i.e., the likelihood of converging to the optimal solution. During each experiment, the host network and its available flexible demand were the same. Thus no resource contention existed. In this simple setting, the upper bound and the lower bound of Agile Net's performance can be observed. The upper bound is due to resource over-provisioning, in which the aggregator has more than enough resource all the time to mitigate congestion when being incentivized by the DSO agent. The lower bound performance was derived from an aggregator where allocations are not significantly changed when being incentivized by the DSO agent because of lack of flexibility. This case is referred as static.

Fig. 3. Average learning duration in seconds under various reconfiguration.

As shown in Fig. 3, RL w/ init showed almost optimal performance for CM. However, without policy initialization, only around 80% of the simulation window Agile Net converges to the optimal decision to the requests of CM within 10 sec. However, more than 15% requests would have an average

TABLE I
OVERALL PERFORMANCE SUMMARY

Performance Measures	Base case	w/o RL	RL w/ init	RL w/o init
Max. congestion duration (hrs)	3.75	2.25	0.75	1.50
Duration %age in emergency	12.3%	10.9%	6.4%	6.7%
Duration %age in alert	55.7%	31.7%	49.1%	46.3%
Incentivization failure rate	—	1.6%	1.2%	1.3%
Incentivization (€/MWh)	—	36.30	64.47	60.68
Max. Net Demand (p.u.)	1.28	1.30	1.23	1.25
Max. Net Generation (p.u.)	0.96	0.74	0.52	0.51
Total Consumption (MWh)	37.42	37.39	37.31	37.03

learning duration larger than 20 sec. Although started with poor policies, the Agile Net was able to adapt to good policies quickly and maintained the performance at a stable level. We attribute the good performance to the highly efficient representation of the Q table. The adaptive tile coding enhanced Q table that was able to generalize to the continuous state space with a limited number of interactions.

2) *Effectiveness*: The overall performance of Agile Net is evaluated through eight parameters, as shown in table I. For an overall performance analysis, simulations are performed for a time window of a year. These performance indicators reveal the efficiency of the approach for mitigating the congestion while maintaining the reliability of supply.

Fig. 4. Yearly load duration curve. Right below figure zoomed in the part (1) of the figure. Left below figure zoomed in the part (2) of the figure.

RL-based incentivization realizes a significant reduction in the duration of the congestion with a slight drop in the total energy consumption. The experiment is simulated in the real-time environment, so it can be presumed that the part of the dispatchable load is still available for future demand dispatch. On the other hand, the allocation of flexible demand results in a high consumption peak in case of incentivization without any learning.

Moreover, RL w/ init reduces the congestion duration almost 80% and 67% concerning the base and w/o learning case, respectively. It is also evident that in both RL based cases, the maximum loading is lower than 1.25p.u. (i.e., 500kVA) and the maximum generation is around 0.5p.u. It is because DSO agent of Agile Net tries to utilize most of the flexibility to mitigate the congestion, thus results in reducing available demand flexibility for future use.

Moreover, the host network experiences the alert situation around 15% more than w/o learning case. Although peak load, as well as the congestion duration, are largely reduced in both RL-based cases, the rebound effect of the demand results in an alert situation. However, the percentages when the network experiences the alert situation in all cases are still found lower than the base case. The overall effect of the methods is depicted in the yearly load duration curve in Fig. 4. The adjusted network load profiles demonstrate a considerable shift from the base case profile. However, loads are shifted mainly in time to keep the energy consumption in the same order.

Fig. 5. Maximum absolute power per day through the transformer during the different seasons of a year.

In Fig. 5, the maximum absolute power through the transformer per day of the year is clustered into four different seasons (i.e., Spring, Summer, Autumn, and Winter). It can be observed that every day there was an instance of congestion through the year in the base case and w/o RL case. On the other hand, maximum absolute power flow through the transformer in the RL based cases are much more scattered.

Further, throughout each season they had some days without any congestion. In summer, there is relatively lower number of days with congestion due to the high availability of solar energy. However, every season has some intrinsic effects on other seasons because the load is being dispatched in real time. For instance, RL w/o init case encounters more days with congestion in autumn than winter. In contrary, RL w/ init encounters more days with congestion in winter. Moreover, concerning w/o RL case, both RL based cases have shown inverse behavior in winter.

IV. CONCLUSION

In this work, an agile concept that allows self-adaptive congestion control and management in an active distribution network is proposed. In the proposed model-free method, the adaptive tile coding serves as a function approximator for a reinforcement learning. Moreover, the brain of Agile Net is the percept agent, which estimates the demand flexibility across the network by developing a responsive virtual environment in light of the price elasticity of demand. The proposed perception technique is used for the first time in this work.

The simulations of Agile Net under different scenarios result in near-optimal incentivization, nevertheless there are still several limitations of this work. First, the incentivization is discrete and in a relatively coarse granularity. Second, the overall performance of Agile Net suffers from initialization considerably.

Future work can extend Agile Net by combining power quality issues within the proposed framework, and formulation of a problem for over/under voltage mitigation using machine learning. It offers further opportunities for the control theory to provide fine-grained operations and stable initial performance.

REFERENCES

- [1] Z. Liu, Q. Wu, S. Oren, S. Huang, R. Li, and L. Cheng, "Distribution locational marginal pricing for optimal electric vehicle charging through chance constrained mixed-integer programming," *IEEE Transactions on Smart Grid*, 2016.
- [2] N. O'Connell, Q. Wu, J. Østergaard, A. H. Nielsen, S. T. Cha, and Y. Ding, "Day-ahead tariffs for the alleviation of distribution grid congestion from electric vehicles," *Electric Power Systems Research*, vol. 92, pp. 106–114, 2012.
- [3] R. A. Verzijlbergh, L. J. De Vries, and Z. Lukszo, "Renewable energy sources and responsive demand. do we need congestion management in the distribution grid?" *IEEE Transactions on Power Systems*, vol. 29, no. 5, pp. 2119–2128, 2014.
- [4] S. Hanif, T. Massier, H. B. Gooi, T. Hamacher, and T. Reindl, "Cost optimal integration of flexible buildings in congested distribution grids," *IEEE Transactions on Power Systems*, vol. 32, no. 3, pp. 2254–2266, 2017.
- [5] P. B. Andersen, J. Hu, and K. Heussen, "Coordination strategies for distribution grid congestion management in a multi-actor, multi-objective setting," in *Innovative Smart Grid Technologies (ISGT Europe), 2012 3rd IEEE PES International Conference and Exhibition on*. IEEE, 2012, pp. 1–8.
- [6] J. Hu, S. You, M. Lind, and J. Østergaard, "Coordinated charging of electric vehicles for congestion prevention in the distribution grid," *IEEE Transactions on Smart Grid*, vol. 5, no. 2, pp. 703–711, 2014.
- [7] B. Biegel, P. Andersen, J. Stoustrup, and J. Bendtsen, "Congestion management in a smart grid via shadow prices," *IFAC Proceedings Volumes*, vol. 45, no. 21, pp. 518–523, 2012.
- [8] C. Zhang, D. Yi, N. C. Nordentoft, P. Pinson, and J. Østergaard, "Flech: A danish market solution for dso congestion management through der flexibility services," *Journal of Modern Power Systems and Clean Energy*, vol. 2, no. 2, pp. 126–133, 2014.
- [9] S. S. Torbaghan, N. Blaauwbroek, P. Nguyen, and M. Gibescu, "Local market framework for exploiting flexibility from the end users," in *European Energy Market (EEM), 2016 13th International Conference on the*. IEEE, 2016, pp. 1–6.
- [10] J. Hu, G. Yang, H. W. Bindner, and Y. Xue, "Application of network-constrained transactive control to electric vehicle charging for secure grid operation," *IEEE Transactions on Sustainable Energy*, vol. 8, no. 2, pp. 505–515, 2017.
- [11] A. Haque, P. Nguyen, F. Bliet, and J. Slooetweg, "Demand response for real-time congestion management incorporating dynamic thermal overloading cost," *Sustainable Energy, Grids and Networks*, vol. 10, pp. 65–74, 2017.
- [12] W. Liu, Q. Wu, F. Wen, and J. Østergaard, "Day-ahead congestion management in distribution systems through household demand response and distribution congestion prices," *IEEE Transactions on Smart Grid*, vol. 5, no. 6, pp. 2739–2747, 2014.
- [13] M. Babar, P. Nguyen, V. Cuk, I. Kamphuis, M. Bongaerts, and Z. Hanzelka, "The evaluation of agile demand response: An applied methodology," *IEEE Transactions on Smart Grid*, 2017.
- [14] L. Hurtado, J. Rhodes, P. Nguyen, I. Kamphuis, and M. Webber, "Quantifying demand flexibility based on structural thermal storage and comfort management of non-residential buildings: A comparison between hot and cold climate zones," *Applied Energy*, vol. 195, pp. 1047–1054, 2017.
- [15] A. Chiş, J. Lundén, and V. Koivunen, "Reinforcement learning-based plug-in electric vehicle charging with forecasted price," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 5, pp. 3674–3684, 2017.
- [16] A. Arif, M. Babar, T. I. Ahamed, E. Al-Ammar, P. Nguyen, I. R. Kamphuis, and N. Malik, "Online scheduling of plug-in vehicles in dynamic pricing schemes," *Sustainable Energy, Grids and Networks*, vol. 7, pp. 25–36, 2016.
- [17] E. Mocanu, P. H. Nguyen, M. Gibescu, and W. L. Kling, "Deep learning for estimating building energy consumption," *Sustainable Energy, Grids and Networks*, vol. 6, pp. 91–99, 2016.
- [18] F. Ruelens, B. J. Claessens, S. Vandael, B. De Schutter, R. Babuška, and R. Belmans, "Residential demand response of thermostatically controlled loads using batch reinforcement learning," *IEEE Transactions on Smart Grid*, 2017.
- [19] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press Cambridge, 1998, vol. 1, no. 1.
- [20] P. A. Centolella, "A pricing strategy for a lean and agile electric power industry," *Electricity Policy*, 2012.
- [21] P. Kundur, N. J. Balu, and M. G. Lauby, *Power system stability and control*. McGraw-hill New York, 1994, vol. 7.
- [22] M. Babar, P. Nguyen, V. Cuk, and I. Kamphuis, "The development of demand elasticity model for demand response in the retail market environment," in *PowerTech, 2015 IEEE Eindhoven*. IEEE, 2015, pp. 1–6.
- [23] S. D. Maqbool, M. Babar, and E. A. Al-Ammar, "Effects of demand elasticity and price variation on load profile," in *Innovative Smart Grid Technologies-Middle East (ISGT Middle East), 2011 IEEE PES Conference on*. IEEE, 2011, pp. 1–5.
- [24] G. Tesaro, N. K. Jong, R. Das, and M. N. Bennani, "On the use of hybrid reinforcement learning for autonomic resource allocation," *Cluster Computing*, vol. 10, no. 3, pp. 287–299, 2007.
- [25] F. L. Lewis and D. Liu, *Reinforcement learning and approximate dynamic programming for feedback control*. John Wiley & Sons, 2013, vol. 17.
- [26] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *International Conference on Learning Representations*, 2016.
- [27] M. Hausknecht, W.-K. Li, M. Mauk, and P. Stone, "Machine learning capabilities of a simulated cerebellum," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 3, pp. 510–522, 2017.
- [28] M. Nijhuis, M. Gibescu, and J. Cobben, "Bottom-up markov chain monte carlo approach for scenario based residential load modelling with publicly available data," *Energy and Buildings*, vol. 112, pp. 121–129, 2016.