Adapting Surprise Minimizing Reinforcement Learning Techniques for Transactive Control

1. Department of Electrical Engineering and Computer Sciences, University of California, Berkeley

ABSTRACT

Optimizing prices for energy demand response requires a flexible controller with ability to navigate complex environments. We propose a reinforcement learning controller with surprise minimizing modifications in its architecture. We suggest that surprise minimization can be used to improve learning speed, taking advantage of predictability in peoples' energy usage. Our architecture performs well in a simulation of energy demand response. We propose this modification to improve functionality and savin a large-scale experiment.

CCS CONCEPTS

• **Computing methodologies** → **Reinforcement learning**; • **Hardware** → **Smart grid**.

1 INTRODUCTION

The electricity grid may be seen as a beautifully decentralized organism: one in which individual energy demands are met by individual generators largely without direct coordination or knowledge of recipient. Market signals simply translate into generators commanding their resources to increase or decrease. However, as volatile resources like wind and solar replace on-demand resources like most fossil fuels – an important development in the drive to decarbonize the energy supply and tackle climate change – a potentially worrying question arises: what happens to demand when the generation becomes decoupled from commands; i.e., when energy is demanded but the sun isn't shining? Grids that do not adequately prepare for this question will face daunting consequences, ranging from curtailment of resources [21] to voltage instability and physical damage.

A common solution is demand response: a strategy in which customers are incentivized to defer loads to periods of the day where energy is plentiful. Given the lack of material infrastructure and cheapness of the incentives, it has several positives above physical energy storage systems. One primary application of demand response is in buildings, and both central-level appliance coordination and building-level demand response has been thoroughly studied in residential and industrial settings ([2], [15], [12], [26], [7].) However, while physical infrastructure of office buildings has been studied for demand response ([8]), there has been no large scale experiment aimed to elicit a behavioral demand response.

The lack of experiment is perhaps understandable when we consider that the majority of offices do not have a mechanism to pass energy prices onto office workers[5]. If they did, however, not only could a large fleet of decentralized batteries – laptops, cell phone chargers, etc. – be coordinated to function as a large deferable resource, but building managers could save money[4].

The SinBerBEST collaboration has developed a Social Game that facilitates workers to engage in competition around energy [10], [11]. Through this framework, a first-of-its-kind experiment has been proposed to implement behavioral demand response within an office building [21]. Prior work has proposed to describe an hourly price-setting controller that learns how to optimize its prices [20].

However, given the costliness of iterations in this experiment, further work in simulation is needed to refine a controller that can adapt the most quickly to the complex behaviors of office workers.

We endeavor to report one such refinement. We will in Section 2 contextualize the architecture of our reinforcement learning controller within the general domain of reinforcement learning and introduce a suprise minimizing algorithm that can further improve this controller. In Section 3 we will describe the simulation setup and the specific modification we propose. In Section 4 we will give results. Finally, in Section 5 we will discuss implications of the controller and the future work this entails.

2 BACKGROUND

2.1 Reinforcement Learning

Reinforcement learning (RL) is a type of agent-based machine learning where control of a complex system requires actions that maximize the agent's outcome [23], i.e. they seek to optimize the expected sum of rewards for actions (a_t) and states (s_t) with a policy p_π and reward given by r in a policy parametirized by θ; i.e., $J(\theta) = \mathbb{E} \sum_{s_t, a_t \sim p_\pi} [r(s_t, a_t)]$. Classical RL saw early use cases in backgammon [24], the cart-pole problem, and Atari [17].

Policy gradient methods are a class of RL algorithms used to train policy networks that suggest actions. We propose the use of a variant of these methods, Proximal Policy Optimization (PPO) [19], which works by computing an estimator of the policy gradient and plugging it into a stochastic gradient ascent algorithm.

RL has been applied to a number of demand response situations, but almost all the work centers on agents that directly schedule resources [14], [13], [27], [25], [18], [6]. RL architectures can vary widely, for example Kofinas et. al. deploys a fuzzy Q-learning multi-agent that learns to coordinate appliances to increase reliability [9]. In another illustrative example, Mbuwir et. al. manages a battery directly using batch Q-learning [16].

2.2 Surprise Minimizing Reinforcement Learning

It is desireable for both the human subjects and the agent that some stability of control is encouraged. Incentivizing the agent to minimize the surprise it experiences is equivalent to incentivizing it to minimize peoples' change in energy usage across time. Forcing the agent to do so would force it to learn a strategy. This corresponds to adjusting people's habits in a stable system rather than forcing them to confront and attempt to undestand an unstable one. Additionally, people behave predictably on aggregate, and thus choosing to minimize surprise may in fact make it easier for the agent to learn.

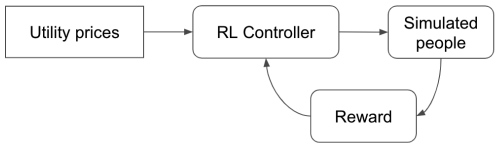

Figure 1: Reinforcement Learning Control Flow

Surprise Minimizing Reinforcement Learning (SMiRL) is an algorithm that aims to reduce the entropy of visited states. SMiRL is useful when the environment provides sufficient unexpected and novel events for learning where the challenge for the agent is to maintain a steady equilibrium state. [3]

SMiRL maintains a distribution $p_\theta(s)$ about which states are likely under its current policy. The agent then modifies its policy π so that it encounters states s with high $p_\theta(s)$, as well as to seek out states that will change the model $p_\theta(s)$ so that future states are more likely.

We make use of SMiRL as an auxiliary reward in addition to our usual reward to calculate a combined reward

$$r_{\text{combined}} = r_{\text{energy}} + \alpha r_{\text{SMiRL}}$$

With a SMiRL weight α as a measure of how much the SMiRL reward influences the total reward. We will describe the explicit formulation of the SMiRL reward in Section 3.

3 METHODS

We adapt SMiRL to the problem of optimizing a price-serving agent for energy demand response. We will briefly explain the baseline Proximal Policy Optimization (PPO). We will then describe the simulation environment we test this in.

3.1 Environment

We summarize an OpenAI gym environment modeled after an environment to simulate demand response in office buildings [22]. Each step in the environment is a day, where the agent proposes prices to office workers, who modify their energy consumption behaviors in order to achieve the lowest cost of energy possible, which the controller then takes as their respective rewards. Each simulated person has a deterministic response to the points that are offered to them.

Can an RL agent learn to provide optimal price signals by implicitly predicting causal factors? Our search space is simple enough to efficiently cover with an entropy maximizing agent, and so we employ a Proximal Policy Optimization (PPO) architecture. We use Ray's RLLib PPO algorithm. The reward for the price-setting agent is

$$r_{\text{energy}} = \log(d^t g)$$

where d is the demand of the person it studies, and g is the grid pricing. We use parameters of learning rate $\alpha = 0.003$, batch size of 256, training stochastic gradient descent minibatch of 32, clip parameter 0.3 and all other parameters were RLLib PPO defaults. For other implementation choices, please see our Github[1]

3.2 SMiRL Reward Formulation

A SMiRL agent recieves an auxiliary reward for experiencing familiar states based on an updating distribution of states it has experienced. This is exactly equivalent to learning a policy with the lowest entropy. Assuming we have a fully-observed controlled Markov process (CMP) with state s_t and action a_t at time t, and $p(s_0)$ as the initial state distribution, and transition probabilities $T(s_{t+1}|s_t, a_t)$, the agent learns a policy $\pi_\phi(a|a)$ parameterized by ϕ. As described earlier, we keep track of an estimated state marginal $p_{\theta_{t-1}}(s_t)$ for the actual state marginal $d^{\pi_\phi}(s_t)$. As usual we denote entropy of a state s_t by $\mathcal{H}(s_t)$. The entropy can then be calculated by the marginal as

$$\sum_{t=0}^{T} \mathcal{H}(\mathbf{s}_t) = -\sum_{t=0}^{T} \mathbb{E}_{\mathbf{s}_t \sim d^{\pi}\phi(\mathbf{s}_t)} \left[\log d^{\pi_\phi}(\mathbf{s}_t)\right] \tag{1}$$

$$\leq -\sum_{t=0}^{T} \mathbb{E}_{\mathbf{s}_t \sim d^{\pi}\phi(\mathbf{s}_t)} \left[\log p_{\theta_{t-1}}(\mathbf{s}_t)\right] \tag{2}$$

We bound (1) by the entropy of an estimated marginal $p_{\theta_{t-1}}$ in (2). Minimizing the right side bound is then equivalent to maximizing an RL objective with reward $r_{\text{SMiRL}}(s_t) = \log p_{\theta_{t-1}}(s_t)$ We note that the optimal policy must also consider future changes to $p_{\theta_{t-1}}(s_t)$ since the distribution of visited states changes at each step. To account for this we use an augmented MDP that captures this notion. [3] We note that in our implementation of SMiRL, $p_{\theta_t}(s)$ is normally distributed. To construct the augmented MDP we include sufficient statistics for $p_{\theta_t}(s)$ in the state space such as the paramaeters of our normal distribution and the number of states seen so far.

3.3 SMiRL Implementation

SMiRL is simply implemented in our existing OpenAI socialgame environment. We introduce SMiRL into our existing socialgame environment by initializing a buffer that tracks the agent's observed states and computes an estimated state marginal p_{θ_t}. As noted earlier, in the augmented MDP the state space also contains the number of observed states and this information is stored in the buffer as well. At each step in our simulation we add newly observed states to our buffer, and update p_{θ_t}. The agent then adjusts the its policy based on the combined reward.

We notice that since $p_\theta(s)$ is modeled as an independent Gaussian for each dimension (hour) in the observation (consumption for a day), then the SMiRL reward is expressed as

$$r_{\text{SMiRL}}(s_t) = -\sum_i \left(\log \sigma_i + \frac{(s_i - \mu_i)^2}{2\sigma_i^2}\right)$$

where μ_i and σ_i are the sample mean and standard deviation from our state marginal and s_i is the i^{th} feature (i^{th} hour of day) of s [3]. With this formulation we can efficiently calculate the SMiRL reward from our buffer.

[1] Our Github may be found at the following link: https://github.com/Aphoh/temp_tc.

3.4 SMiRL as an Auxilary Reward

We use SMiRL as an auxilary reward to provide faster learning and more stable outputs. We achieve this by calculating the SMiRL reward r_{SMiRL} as described in the previous section, and applying a SMiRL weight α to it and then using the sum with our usual energy reward $r_{\text{energy}} = \log(d^t g)$. This gives us a combined reward of

$$r_{\text{combined}} = r_{\text{energy}} + \alpha r_{\text{SMiRL}} \tag{3}$$

and it is this reward that we use to train the RL agent. In our simulations, we found the optimal SMiRL weight α to be around $\alpha = 0.12$ after hyperparameter tuning. We will discuss the exact results of various SMiRL weights in Section 4.

4 RESULTS

We now discuss our two aims: more effective learning and a more stable environment.

4.1 Sample Entropy

4.1.1 Observing Lower Sample Entropy. As the primary objective of a surprise minimization technique is to, in fact, minimize surprise, it is natural to determine whether SMiRL does so.

We first quantify the degree to which the environment is more or less entropic when under the influence of an agent employing SMiRL. Since we assume each variable in our sample space is normally distributed in a given timeframe, we can compute its entropy as $H = \frac{1}{2}\ln(2\pi e\sigma^2)$. For each step, we compute the sample variance of the last 100 steps and use this to compute a sample entropy, shown in **Figure 2**.

While both agents do reduce the sample entropy over time, the SMiRL + PPO agent does so earlier than the baseline PPO agent, with significant decay beginning around step 50k v.s. step 100k. The SMiRL + PPO implementation exhibits lower sample entropy for all iterations compared to baseline PPO, and hence energy usage over the course of the simulation is more consistent when using SMiRL.

Additionally, the SMiRL + PPO agent converges towards a policy which generates a more stable final environment than the baseline PPO agent. The SMiRL + PPO agent's behaviour confirms our hypothesis that the stability of our environment is superior to our baseline PPO agent by incentivizing the RL agent to revisit familiar states. We also note that the both agents continue to explore changes to their policies after convergence, however the SMiRL + PPO maintains a lower sample entropy in late timesteps. Hence, the SMiRL + PPO agent explores in ways that maintain a more stable environment, while the PPO agent's exploration results in more unstable energy usage.

In this sense, the addition of the SMiRL reward can allow an agent to strike a balance between exploration and stability.

4.2 Improved Learning with SMiRL (Compared to Baseline PPO)

While environment stability is important, it is essenital that the agent still encorages efficient energy usage. We also wish to understand whether the SMiRL reward models our assumption of predictability in our system by improved learning speed.

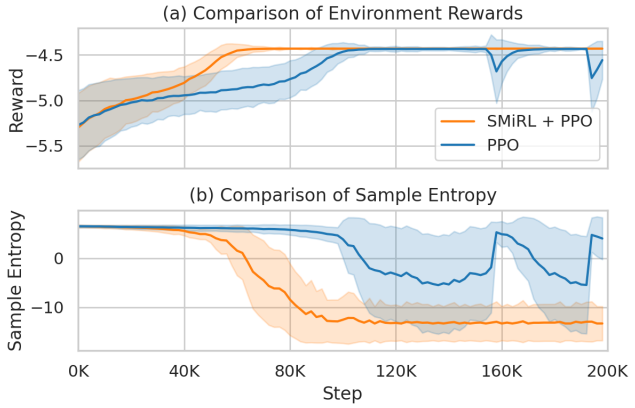

Figure 2: A comparison between the PPO + SMiRL agent and the baseline PPO agents' (a) rewards and (b) sample entropies over training steps. Shaded regions are one standard deviation of observations binned to every 100 steps.

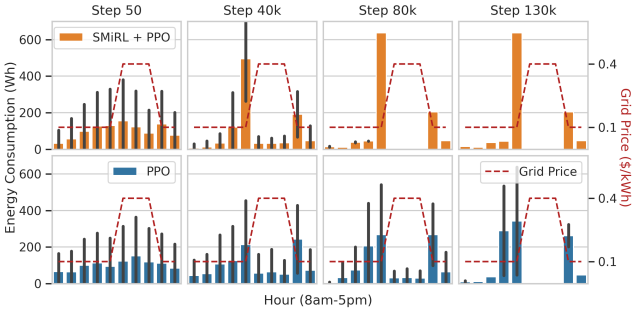

Figure 3: Energy consumption with the PPO and SMiRL + PPO agent at steps 10k, 40k, 80k, and 130k compared to the grid price. Please see section 5 for a brief note on the seemingly adverse behavior in the SMiRL+PPO.

4.2.1 Faster Learning and Consistent Outcomes. We observe that our SMiRL + PPO implementation induces significantly faster learning and convergence to an equally optimal policy, with the same reward. As shown by Figure 2(a), both agents maintain similar rewards up until step ~30k, after which the SMiRL + PPO agent begins to achieve, on average, a higher reward. The SMiRL + PPO implementation converges in roughly half the time as the baseline PPO agent (step ~50k v.s. step ~110k). The reward in the environment around step ~60k whereas the PPO baseline does not converge to the maximum reward in the environment until step ~110k. Our results support our hypothesis that an auxilary SMiRL reward improves learning speed.

Note that we compare the energy rewards (r_{energy}) here directly and not the combined reward which includes the SMiRL reward and weight. This demonstrates the influence of the SMiRL reward on the energy reward, which describes the overall effectiveness of our agent. Hence the inclusion of the SMiRL reward results in improved learning speed of our agent in the task it is given.

Faster convergence is demonstrated by 3, where energy consumption is shown at different steps in the environment. The grid price signal the agent recieves is shown. We observe that, for both the PPO and SMiRL + PPO agent, the price signal effectively shifts people's energy consumption away from times when the grid price is higher. By step 40k, however, the SMiRL + PPO agent has already begun to greatly reduce consumption during peak, and people have shifted it towards just before that peak. By step 80k, the SMiRL + PPO agent has completely diminished any energy usage during the peak pricing, while the PPO agent only does so by step ~120k (when its reward converges).

4.2.2 Optimal SMiRL Weights. We note that while an appropriate SMiRL weight provides significant improvement in learning speed and sample entropy, an inadequate SMiRL weight can lead to poor learning that converges to a suboptimal result, and may not outperform a baseline PPO.

Specifically, we found SMiRL weights of $\alpha = 0.25$ and higher performed worse than baseline PPO. In fact, we find that they converge to a suboptimal reward in the environment, hinting that too much surprise minimization might hinder exploration.

For much lower SMiRL weights such as $\alpha = 0.01$, we do not see any significant benefits when compared to baseline PPO; there isn't enough weight on the SMiRL reward to have a meaningful impact.

4.2.3 Sample Entropy and Environment Reward Curves. When comparing the sample entropy and reward curves in **Figure 2** we see that beginning at steps ~50k, the SMiRL + PPO agent's observed sample entropy drops significantly below that of the baseline PPO agent. This correlates closely at ~50k steps where the PPO + SMiRL implementation begins to experience significantly greater rewards than baseline PPO. Lastly, we observe that at ~110k steps, both agents have observed a drop in sample entropy, and their environment rewards converge. This correlation between entropy and reward may support our hypothesis that aiming for a stable environment via surprise minimization can help the agent learn faster.

5 DISCUSSION

RL controllers are particularly well suited for dynamic data driven environments with unknown and changing system models. Whereas a non-RL controller could potentially require consumer models, or try to learn those explicitly, an RL controller can adapt to dynamically changing resources over time. The novelty of our proposition is in using reinforcement learning to preemptively offer optimal pricing without soliciting demand/supply bids, and purely from historical and forecasted data.

A careful observer may note in Figure 3 that although the energy consumptions are certainly minimizing the price that a building manager might pay (i.e. the controller's reward), large peaks would be unfavorable for the grid if incentivized large scale. We argue that this is not a defect perse of the SMiRL modification; only the exploitation of an externality our simulation does not account for. First, the SMiRL agent is achieving a goal: environment stability, evidenced in the tightening of confidence bounds around energy consumption outcomes. Outcome behavior confidence is accomplished by incentivizing people's consumption tightly up against the shoulder of TOU pricing because the mechanics of the simulated office worker cause it to curtail and, notably, *shift* energy when offered high prices. Therefore the controller learns to rely on shifting. While the SMiRL + PPO and PPO agents are equal in the energy reward the controller receives, the inclusion of the SMiRL reward creates more consistent energy usage. The behavior is optimal given the externalities we have encoded (or not encoded) into the reward, so should not be seen as a deficit of the controller. It would be straightforward, and the subject of future work, to modify the reward in order to direct the controller to spread energy evenly amonst non-peak hours.

After consider this, SMiRL has shown promising results in minimizing surprise, and improvements in learning speeds support our hypothesis that SMiRL can take advantage of inherent predictability in people's energy consumption.

From the energy consumptions in Figure 3 we note that SMiRL induces faster convergence to cheaper energy consumptions. For a building that uses a SMiRL based controller, this translates into direct savings in energy cost.

While the goals of SMiRL stand in contrast to proven entropy maximizing RL algorithms, the results of our simulation and related work show that in appropriate environments, the auxiliary SMiRL reward can improve learning while also letting the RL agent explore. We argue that a human-in-the-loop environment is necessarily complex enough that encouraging surprise minimization encourages the agent to more quickly grasp the latent similarities that govern the environment. We look towards our experiment to continue to demonstrate this.

5.1 Future Research

SMiRL can handle variations in a k-discrete state-space by creating k separate buffers, one for each variation of the state-space, that shrink the action sampling towards each k category. We propose to adapt this to a continuous state-space by clustering main categories of observation. One of the main inputs to the state space is price signals, so we propose to cluster based on a semantic feature search proposed in [1] for building load profiles. In this manner, we will chart novel territory in the SMiRL domain and accommodate more complex environments. More practically, we will use this enhancement directly in the rollout of our agent for our office-scale demand response experiment.

5.2 Acknowledgements

We gratefully acknowledge the input of Manan Khattar, Austin Jang, and Dustin Luong as working group collaborators who met weekly on their own thrusts and provided advice (and at times a listening ear.) We give thanks to Andreea Bobu and Peter Hendrickson who are graduate students that gave guidance during this process. Finally, we would like to thank Glen Berseth for his initial work on SMiRL and Pieter Abbeel and Sergey Levine for advice throughout the process.

REFERENCES

[1] Milad Afzalan and Farrokh Jazizadeh. 2019. Semantic search in household energy consumption segmentation through descriptive characterization. In *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation.* 263–266.

[2] Ailin Asadinejad, Alireza Rahimpour, Kevin Tomsovic, Hairong Qi, and Chien-fei Chen. 2018. Evaluation of residential customer elasticity for incentive based demand response programs. *Electric Power Systems Research* 158 (2018), 26–36.

[3] Glen Berseth, Daniel Geng, Coline Devin, Chelsea Finn, Dinesh Jayaraman, and Sergey Levine. 2019. SMiRL: Surprise Minimizing RL in Dynamic Environments. *CoRR* abs/1912.05510 (2019). arXiv:1912.05510 http://arxiv.org/abs/1912.05510

[4] Hari Prasanna Das, Ioannis Konstantakopoulos, Aummul Baneen Manasawala, Tanya Veeravalli, Huihan Liu, and Costas J Spanos. 2020. Do Occupants in a Building exhibit patterns in Energy Consumption? Analyzing Clusters in Energy Social Games. (2020).

[5] Hari Prasanna Das, Ioannis C Konstantakopoulos, Aummul Baneen Manasawala, Tanya Veeravalli, Huihan Liu, and Costas J Spanos. 2019. A novel graphical lasso based approach towards segmentation analysis in energy game-theoretic frameworks. In *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*. IEEE, 1702–1709.

[6] Danilo Fuselli, Francesco De Angelis, Matteo Boaro, Stefano Squartini, Qinglai Wei, Derong Liu, and Francesco Piazza. 2013. Action dependent heuristic dynamic programming for home energy resource scheduling. *International Journal of Electrical Power & Energy Systems* 48 (2013), 148–160. https://doi.org/10.1016/j.ijepes.2012.11.023

[7] Brandon J Johnson, Michael R Starke, Omar A Abdelaziz, Roderick K Jackson, and Leon M Tolbert. 2015. A dynamic simulation tool for estimating demand response potential from residential loads. In *2015 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*. IEEE, 1–5.

[8] Y. Kim. 2018. Optimal Price Based Demand Response of HVAC Systems in Multizone Office Buildings Considering Thermal Preferences of Individual Occupants Buildings. *IEEE Transactions on Industrial Informatics* 14, 11 (2018), 5060–5073. https://doi.org/10.1109/TII.2018.2790429

[9] P. Kofinas, A.I. Dounis, and G.A. Vouros. 2018. Fuzzy Q-Learning for multi-agent decentralized energy management in microgrids. *Applied Energy* 219 (2018), 53–67. https://doi.org/10.1016/j.apenergy.2018.03.017

[10] Ioannis C Konstantakopoulos, Andrew R Barkan, Shiying He, Tanya Veeravalli, Huihan Liu, and Costas Spanos. 2019. A deep learning and gamification approach to improving human-building interaction and energy efficiency in smart infrastructure. *Applied energy* 237 (2019), 810–821.

[11] Ioannis C Konstantakopoulos, Hari Prasanna Das, Andrew R Barkan, Shiying He, Tanya Veeravalli, Huihan Liu, Aummul Baneen Manasawala, Yu-Wen Lin, and Costas J Spanos. 2019. Design, benchmarking and explainability analysis of a game-theoretic framework towards energy efficiency in smart infrastructure. *arXiv preprint arXiv:1910.07899* (2019).

[12] Chaojie Li, Chen Liu, Xinghuo Yu, Ke Deng, Tingwen Huang, and Liangchen Liu. 2018. Integrating Demand Response and Renewable Energy In Wholesale Market.. In *IJCAI*. 382–388.

[13] D. Li and S. K. Jayaweera. 2014. Reinforcement learning aided smart-home decision-making in an interactive smart grid. In *2014 IEEE Green Energy and Systems Conference (IGESC)*. 1–6. https://doi.org/10.1109/IGESC.2014.7018632

[14] Y. Liu, C. Yuen, N. Ul Hassan, S. Huang, R. Yu, and S. Xie. 2015. Electricity Cost Minimization for a Microgrid With Distributed Energy Resource Under Different Information Availability. *IEEE Transactions on Industrial Electronics* 62, 4 (2015), 2571–2583. https://doi.org/10.1109/TIE.2014.2371780

[15] Kai Ma, Guoqiang Hu, and Costas J Spanos. 2015. A cooperative demand response scheme using punishment mechanism and application to industrial refrigerated warehouses. *IEEE Transactions on Industrial Informatics* 11, 6 (2015), 1520–1531.

[16] Brida V. Mbuwir, Frederik Ruelens, Fred Spiessens, and Geert Deconinck. 2017. Battery Energy Management in a Microgrid Using Batch Reinforcement Learning. *Energies* 10, 11 (2017). https://www.mdpi.com/1996-1073/10/11/1846

[17] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. 2013. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602* (2013).

[18] Leo Raju, Sibi Sankar, and R.S. Milton. 2015. Distributed Optimization of Solar Micro-grid Using Multi Agent Reinforcement Learning. *Procedia Computer Science* 46 (2015), 231–239. https://doi.org/10.1016/j.procs.2015.02.016 Proceedings of the International Conference on Information and Communication Technologies, ICICT 2014, 3-5 December 2014 at Bolgatty Palace & Island Resort, Kochi, India.

[19] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal Policy Optimization Algorithms. *CoRR* abs/1707.06347 (2017). arXiv:1707.06347 http://arxiv.org/abs/1707.06347

[20] Lucas Spangher, Akash Gokul, Manan Khattar, Joseph Palakapilly, Utkarsha Agwan, Akaash Tawade, and Costas Spanos. 2020. Augmenting Reinforcement Learning with a Planning Model for Optimizing Energy Demand Response. In *Proceedings of the 1st International Workshop on Reinforcement Learning for Energy Management in Buildings & Cities*. 39–42.

[21] Lucas Spangher, Akash Gokul, Manan Khattar, Joseph Palakapilly, Akaash Tawade, Adam Bouyamourn, Alex Devonport, and Costas Spanos. 2020. Prospective experiment for reinforcement learning on demand response in a social game framework. In *Proceedings of the Eleventh ACM International Conference on Future Energy Systems*. 438–444.

[22] Lucas Spangher, Akash Gokul, Joseph Palakapilly, Utkarsha Agwan, Manan Khattar, Wann-Jiun Ma, and Costas Spanos. [n.d.]. OfficeLearn: An OpenAI Gym Environment for Reinforcement Learning on Occupant-Level Building's Energy Demand Response. In *Tackling Climate Change with Artificial Intelligence Workshop at NeurIPS, 2020*.

[23] Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

[24] Gerald Tesauro. 1994. TD-Gammon, a self-teaching backgammon program, achieves master-level play. *Neural computation* 6, 2 (1994), 215–219.

[25] Q. Wei, D. Liu, and G. Shi. 2015. A novel dual iterative Q-learning method for optimal battery management in smart residential environments. *IEEE Transactions on Industrial Electronics* 62, 4 (2015), 2509–2518. https://doi.org/10.1109/TIE.2014.2361485

[26] Ji Hoon Yoon, Ross Baldick, and Atila Novoselac. 2014. Dynamic demand response controller based on real-time retail price for residential buildings. *IEEE Transactions on Smart Grid* 5, 1 (2014), 121–129.

[27] Y. Zhang and M. van der Schaar. 2014. Structure-aware stochastic load management in smart grids. In *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*. 2643–2651. https://doi.org/10.1109/INFOCOM.2014.6848212