Energy Storage in the Smart Grid: a Multi-Agent Deep Reinforcement Learning Approach

Pawel Knap[1] and Enrico Gerding[1]
University of Southampton, UK, pmk1g20@soton.ac.uk

Abstract. This paper introduces an energy storage system controlled by a reinforcement learning agent for smart grid households. It optimizes electricity trading in a variable tariff setting, yielding consumer savings averaging 20.91% annually without altering consumption habits. Integrated with solar panels, it offers even greater cost reductions. A Multi-Agent System simulation analyzes interactions between agents and identifies beneficial price-demand relationships. Moreover, it shows storage's positive impact on the energy market for operators and consumers. Deep Q Learning is identified as the most effective algorithm, efficiently managing high-dimensional, nonstationary, and stochastic aspects of the problem, bypassing the need for abstract modelling and deterministic rules. Furthermore, our ablation study explores various storage sizes and agent complexities.

Keywords: Smart Grid, Deep Reinforcement Learning, Multi-Agent System, Energy Storage

1 Introduction

Our study proposes an energy storage solution controlled by Deep Reinforcement Learning (DRL) to address fluctuating electricity costs in the smart grid (SG). Utilizing real-world data from the Low Carbon London project [22] and Octopus variable tariff data [18], a self-interested DRL agent makes decisions based on price, storage level, and stored electricity value every 30 minutes. The study investigates the concurrent usage of storage and photovoltaic panels (PV), and simulates a community of households to evaluate their behaviour, cooperation-competition patterns, and impact on the power grid. Various agent types, action capabilities, storage capacities, and PV powers are tested. Results indicate significant consumer savings and grid stress reduction. In summary, our study examines the benefits and challenges of SG, highlighting the effectiveness of in-house energy storage controlled by a selfish DRL agent. Full source code and supplementary material are available at https://github.com/PawelKnap/EneStore.

2 Background

In reinforcement learning (RL), an agent aims to optimize its policy to maximize cumulative rewards. The RL cycle, shown in figure 1, involves the agent selecting actions, thus altering the environment state, and receiving rewards. A policy $\pi(a_1, s_1) = Pr(a_1|s_1)$ dictates the agent's actions based on the current environment state. The value function is computed as $V(s_1) = E(\sum_{t=1}^{\infty} \gamma^t r_t|s_1)$, approximating the expected cumulative reward in a given state. Another key concept is the quality function, $Q(s_1, a_1) = E[R(s_2, s_1, a_1) + \xi V(s_2)]$, assessing state-action pairs' quality considering immediate and future rewards. This function operates within a Markov Decision Process (MDP). MDP's crucial feature

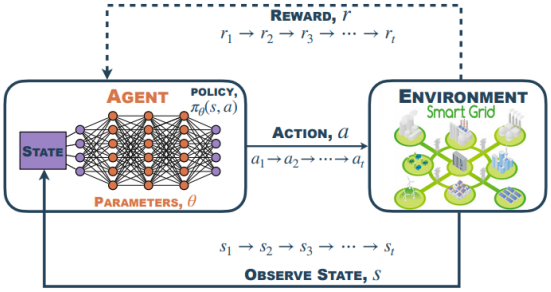

Fig. 1: Reinforcement Learning cycle. An agent in state s_1 chooses action a_1, affecting the environment to transition to s_2 and receiving reward r_1, and the whole process repeats itself. Adapted from [7].

is the conditional independence of the next state from prior states and actions. The value and policy functions can be derived from the Q function, with the former representing maximum value as $V(s_1) = max_a Q(s_1, a)$, and the latter indicating the action with the maximum $V(s_1)$ as $\pi(a_1, s_1) = argmax_a Q(s_1, a)$.

2.1 Used method overview

In Q-learning, an agent updates its Q function iteratively using the formula: $Q^{new}(s_1, a_1) = Q^{old}(s_1, a_1) + \alpha[r_1 + \gamma max_a Q(s_2, a) - Q^{old}(s_1, a_1)]$, where α is the learning rate. Here, $r_1 + \gamma max_a Q(s_2, a)$ represents the target estimate of cumulative reward, serving as part of an error signal which guides the Q function update, with zero error indicating optimal policy discovery. The update increases the Q value if the agent's reward exceeds expectations, or decreases it otherwise. In practice, the Q function is a table of Q values for different states and actions.

Deep Q Learning (DQL) [17], an off-policy gradient-free technique, employs deep neural networks (NN) to approximate the Q function for decision-making. While very deep NNs excel in tasks like Computer Vision [13], DRL often opts for simpler NN architectures with 2-3 hidden layers [24,27]. During training, it balances exploration and exploitation through trial-and-error, gradually transitioning from random to informed actions based on experience, focusing exploration on promising paths. In DQL, the quality function $Q(s_1, a_1)$ is parameterized by NN weights Θ, addressing high-dimensional feature approximation and the curse of dimensionality [6]. The NN minimizes the squared error expectation via gradient descent and backpropagation to optimize parameters Θ, determining the optimal Q-function. Experience replay [16] enhances generalization using previous transitions to update NN weights. Additionally, two NNs, the policy and target network, manage correlations and improve stability.

3 Report of literature search

Whilst there is agreement that storage presence in SG is beneficial, optimal deployment location remains debated. Barbour et al. [4] advocate for community batteries over individual household storage to reduce energy exchange with the grid. However, their approach increases demand peaks, adding to grid stress.

Voice et al. [25] propose decentralized control of micro-storage selfish agents in MAS, reducing supplier costs by 16% and enhancing robustness, yet constant daily load and equal import-export price assumptions may limit applicability. Their approach also overlooks the self-use of stored energy and renewable integration. Wang et al. [26] suggest shared ownership of household storage, improving peak reduction and network investment, but our approach offers similar advantages while being simpler. Furthermore, their joint storage control raises privacy concerns. Deng et al [8] propose price schemes to influence consumption patterns thus reducing demand peaks, whereas our solution achieves the same without requiring behaviour changes.

An overview of energy marketplace models and dynamic pricing techniques for SG is provided in [15,5,14]. A recent study [11] introduces an RL-based energy market model for prosumer-dominated microgrids. Employing multi-agent reinforcement learning (MARL), it establishes a dynamic pricing environment linked to real-time demand, resulting in increased profits for prosumers and the grid operator, along with reduced grid reserve power utilization. However, it focuses solely on prosumers, neglecting consumers. Another work [21] presents a MARL-based solution for industrial sites to manage electricity costs amid fluctuating prices and growing renewable power generation. This approach optimizes production resources, battery storage, self-generation, and market trading to minimize expenses. Comparative assessments demonstrate the superiority of the MARL system over rule-based strategies in terms of speed and quality. However, it simplifies cooperative agents' behaviours by considering only essential information, potentially overlooking their complete intricacy.

Machine Learning (ML) algorithms have diverse applications in SG. For example, Atef and Eltawil [3] forecast electricity prices, Asare-Bediako et al. [2] use NNs for residential load profile forecasting, and Eck et al. [10] predict local energy demand. Singh et al. [23] propose a novel MAS-based system for load frequency control, outperforming previous algorithms, leveraging distributed RL controllers and swarm optimization. Additionally, Qiu et al. [20] explore DRL's approach to peer-to-peer energy trading, offering an alternative SG functioning method. Ali and Choi [1] provide a comprehensive review of ML techniques in SG, highlighting six main development areas and discussing market liberalization and economic aspects.

4 Experiment setup

The experiment used electricity consumption data from the Low Carbon London project [22], involving 5,567 London households' smart meters data from November 2011 to February 2014. This data was merged with variable tariff prices from Octopus Energy [18], resulting in a dataset spanning over 15 million episodes for single-agent simulations. Storage sizes of 0.5kWh, 1.5kWh, and 3kWh were studied, corresponding to available in-house socket powers. Agent operations are shown in Figure 1. Agents make decisions based on the environmental state in each half-hour episode, with actions including waiting, purchasing, using, or selling energy. The range of environment states varies depending on the scenario, from 110 in single-agent to infinity in MAS simulations.

4.1 Simulation of solar panels energy generation

To model households as prosumers, we employed an algorithm simulating PV energy generation due to insufficient real data for the long training period. For each interval excluding nighttime, energy production in kWh was computed. Daytime duration, average sunrise, and sunset times were calculated for London, remaining constant for 30 days, resulting in a 360-day simulation year. Solar power production patterns were simulated using Gaussian functions with small random variations. Additionally, brief periods of low energy production due to cloud cover were simulated by adjusting energy production with probabilities. Figure 2c illustrates a typical solar generation pattern. The energy within each interval is scaled based on the average daily energy production, which varies monthly. Simulations can be repeated for any number of years, with each daily generation pattern being unique. The simulated annual energy generation varies between 800 and 900 kWh for a 1kW system (proportionally more for higher powers), as detailed in [9].

4.2 Single-agent simulation

Baseline algorithm operates on a simple rule: it buys energy for storage when the price is below 9 p/kWh, uses stored energy when the price exceeds 18 p/kWh, and waits otherwise. Any electricity deficit is bought from the grid. In the prosumer simulation, all generated energy is immediately sold.

Deep Q Learning agent employs the reward policy detailed in Algorithm 1. Epsilon linearly decreases from 1 to 0 over 15 million training examples, while gamma is set to 0. A replay memory of 1024 episodes updates a neural network with one hidden layer of 2048 units, using a batch size of 32 examples and the Adam optimizer with a learning rate of 10^{-9}. The target network updates every 1000 episodes. Similar to the baseline, the agent chooses from three actions (buy, use, wait) in each time interval, purchasing any deficit from the grid. The agent's environment state is defined by storage level, stored energy value, and current electricity price. To optimize results, the latter two variables are binned into 10 categories based on price per kWh, while storage filling is binary (empty or not), resulting in 110 states due to the correlation between storage filling level and stored energy value (which is 0 when storage is empty).

DQL agent with increased action space. Exploring the addition of a fourth action allowing agents to sell stored energy aimed to boost savings in scenarios with full storage, high export prices, and low energy consumption, or when export prices exceed import prices. The NN incorporates the selling price as a fourth input, modelled by the Octopus export tariff binned similarly to import prices (10 bins each). Thus, the state space increases from 110 to 1,100. The reward policy, including the red colour code, is detailed in Algorithm 1. All other parameters remain unchanged from subsection 4.2.

DQL agent as a single prosumer is created by integrating the PV simulator into the 3-action DQL agent without retraining. Octopus's export tariff determines the sell price. The produced energy is firstly used for household consumption, with surplus either stored or sold if storage is full. Alternatively, generated

energy is added to storage or sold if storage space is insufficient. The former excelled in single-agent, while the latter performed better in MAS simulations, as explained in the Appendix.

4.3 MAS simulation

3-action DQL agents were combined in a MAS to simulate a community of three households, with each represented by one retrained agent. They operated in a shared electricity market with prices determined by community demand. Various price-demand functions, including linear, logarithmic, and exponential, were modelled as follows: $Price = 3 \times demand, Price = 11 \times ln(3 \times demand + 0.7), Price = exp(demand/11.25) - 1$. The decision to include the logarithmic function was influenced by findings from Lipman [12], showing a logarithmic relationship between net demand and price in the Octopus tariff. Agents underwent retraining with 1.2 million episodes in the MAS scenario, maintaining a reward policy as described in section 4.2. Threshold price values for purchasing decisions were 3.075, 9.04, and 0.19 p/kWh for linear, logarithmic, and exponential functions. They were chosen based on the observation that around 60% of instances in the training set had prices lower than these thresholds. The same principle guided the setting of the original threshold values.

Agent's NNs have two hidden 2048-wide layers, with continuous input data on price, stored energy quantity, and its value. Distinct 128-wide replay memories provide training batches of 32 samples, with target networks updated every 100 episodes. Adam optimizers with a learning rate of 10^{-6} were employed. Gamma was set to 0, and epsilon decreased linearly from 1 to 0. Testing was conducted on three datasets, each with 300,000 episodes per agent. In the MAS with the logarithmic price function, an adjustment was made to accommodate negative prices, with such purchases receiving a reward of 10. Since the initial price depends on community demand, it's assumed the energy supplier accurately predicts demand for the upcoming interval. However, agents' actions influencing demand cause price variation, posing a challenge as they base decisions on the initial price, different from the final price used for bill calculations. Finally, PV data was included in the MAS simulation with logarithmic prices to model households as prosumers. The selling price is set at 80% of the import price, unless the import price is negative, in which case the selling price is 0, effectively mirroring real-world scenarios due to inherent regularization.

5 Results

5.1 Octopus Agile Import and Export tariff savings

Switching from a fixed tariff to the variable Octopus Agile tariff can yield savings for households, as shown in Table 4. Costs rise only in 1 out of 40 cases due to high peak consumption, but our storage system offsets this loss. Unless specified, all simulations measure savings relative to the Octopus Agile tariff's electricity cost. In the first quarter of 2019, the mid-level PV export tariff was 3.54 p/kWh, according to Ofgem [19]. This rate is used here to demonstrate the advantages of a variable export tariff. Table 5 presents annual profit statistics resulting from the tariff change, compared to earnings under the Octopus Agile export tariff.

5.2 Results of a single agent simulations

Table 1 illustrates the superiority of the DQL approach over the baseline, with a slight improvement for the 4-action DQL, except in the case of a 0.5 kWh storage. The simulations underscore that significant savings stem from combining storage and solar panels, showcasing agents' efficacy as prosumers. However, expanding battery capacity for a given PV power does not substantially increase savings, as evaluated battery sizes are not adequate for simulated PV power generation.

Table 1: Mean yearly savings from various agents with different storage sizes, relative to no storage case. With PV, surplus energy is immediately sold.

	0.5kWh battery		1.5kWh battery		3kWh battery	
Metrics	Savings in £	Savings in %	Savings in £	Savings in %	Savings in £	Savings in %
Baseline	19.26 ± 2.81	6.35 ± 2.79	44.97 ± 12.95	13.39 ± 3.67	61.27 ± 22.2	17.42 ± 3.77
DQL	21.7 ± 3.01	7.13 ± 3.08	51.4 ± 14.92	15.21 ± 4.01	68.91 ± 26.96	19.2 ± 3.76
DQL 4-actions	21.63 ± 2.77	7.01 ± 3.31	51.89 ± 13.31	15.35 ± 5.03	73.23 ± 23.6	20.91 ± 5.76
DQL with 1kW PV	74.17 ± 23.86	24.97 ± 13.62	77.19 ± 24.79	25.94 ± 13.82	78.4 ± 25.94	26.49 ± 14.72
DQL with 4kW PV	275.39 ± 74.72	90.33 ± 45.41	275.53 ± 74.76	90.27 ± 45.4	275.93 ± 74.96	90.47 ± 45.38

5.3 Results of MAS simulations

Table 2 demonstrates varying savings influenced by individual agents' consumption patterns. The exponential function produces the highest savings, followed by the linear function, indicating that steeper price-demand functions correlate with greater savings. Additionally, specific consumption patterns contribute to enhanced savings, with agent 2 achieving the highest savings, with agent 0 ranking second across all price-demand functions. Furthermore, Table 3 displays saving rates for simulations of MAS with solar panels energy generation.

Table 2: Savings as a percentage ratio between costs with and without the 0.5 kWh storage system under different price-demand functions in MAS simulation.

Price-demand function	Agent 0 savings in %	Agent 1 savings in %	Agent 2 savings in %
Logarithmic	1.87 ± 0.56	1.52 ± 0.35	2.52 ± 0.24
Linear	3.16 ± 0.45	2.73 ± 0.38	4.75 ± 0.29
Exponential	5.05 ± 1.57	4.65 ± 0.75	5.67 ± 0.4

Not all consumers may prefer to use storage. Hence, the analysis extends to evaluate the impact of households with storage on the electricity expenses of those without it. The resulting cost reductions for various scenarios and the percentage deviation in savings compared to the scenario where all agents have storage facilities are summarized in Table 6. This table also highlights that agents exert varying influences on neighbourhood savings. The absence of storage in agent 0 slightly affects the savings of all agents, including itself. Agents 1 and 2 experience a decrease of under 3% in scenarios where other agents lack storage. However, if they are without storage, their savings drop by more than 50%. Notably, agent 0 sees a significant decline in savings when any community member

loses access to storage. Nevertheless, each agent saves when at least one of them possesses a storage system.

Table 3: Savings expressed as a percentage ratio between costs with storage and PV versus costs when all agents are without storage, in parentheses without storage and PV. Agents operate in a shared market with logarithmic prices. Cases 1, 2, and 3 involve all agents having storage of 0.5, 1.5, and 3 kWh, coupled with PV of 1, 4, and 6 kW, respectively. In case 4, agent 0 has 0.5 kWh storage and 1 kW PV, agent 1 has 1.5 kWh storage and 4 kW PV, and agent 2 has 3 kWh storage and 6 kW PV. In case 5, agents 0 and 1 lack both storage and PV, while agent 2 has 3 kWh storage and 6 kW PV.

Case	Agent 0 savings in %	Agent 1 savings in %	Agent 2 savings in %
1	2.11 ± 0.57(24.34 ± 2.69)	0.89 ± 0.53 (21.02 ± 3.69)	2.77 ± 1.13 (21.82 ± 1.8)
2	2.04 ± 0.78(43.39 ± 2.33)	1.07 ± 0.83(36.27 ± 6.28)	4.95 ± 0.99(38.59 ± 1.75)
3	1.87 ± 0.9 (46.06 ± 1.83)	-0.2 ± 0.21 (37.85 ± 6.09)	4.88 ± 0.65 (41.3 ± 2.44)
4	2.45 ± 1.32(40.95 ± 2.83)	1.52 ± 0.78 (35.81 ± 6.14)	9.99 ± 1.11(41.54 ± 0.87)
5	1.28 ± 0.67 (28.57 ± 2.6)	0.83 ± 0.31 (23.95 ± 4.1)	9.44 ± 1.97(38.11 ± 0.63)

6 Discussion

6.1 Single agent simulations

Figure 2a illustrates a logarithmic increase in median saving rates with storage size, plateauing around 22% for capacities over 4 kWh. These findings, coupled with the significantly higher costs of high-capacity storage, make the installation of large storage facilities economically unfeasible. Technological limitations also arise, such as the maximum power capacity of home charging points. Furthermore, analysis shows savings decrease as total yearly electricity consumption increases (Figure 2b).

6.2 MAS simulation

The MAS simulation yielded significant findings. Firstly, certain functions, notably the exponential one, offer more benefits to consumers (Table 2). This is because agents utilize stored energy during peak times and buy it during off-peak hours. The steeper price-demand function results in higher price disparities between peak and off-peak periods, leading to greater savings. Secondly, the presence of at least one household with a storage system reduces electricity costs for others without it. However, savings are higher for these households when each has its own storage (Table 3). Thirdly, the agents' behaviour examination reveals no clear signs of cooperation or competition.

Table 3 illustrates that larger battery capacities and more powerful PV systems lead to increased savings compared to households without panels and storage. The values in the table represent savings resulting from all storage within the system. This explains why agents 0 and 1 show non-zero values in case 5, despite lacking storage themselves, and why Agent 1 records negative savings in case 3 due to other agents' storage impact. Reducing storage size and PV power for agents 0 and 1 in case 4 doesn't substantially decrease total savings compared

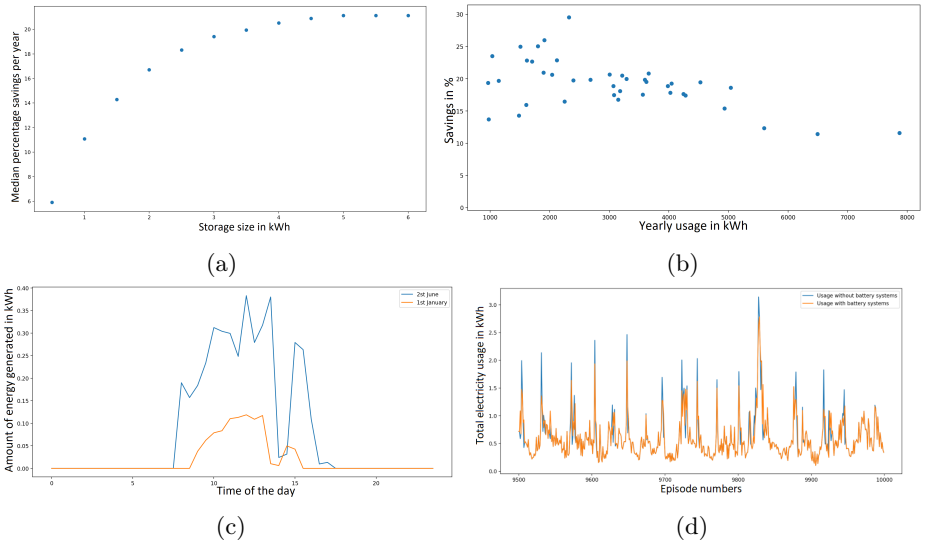

Fig. 2: Median percentage yearly saving of DQL agent versus storage capacities showed in (a), and a comparison of yearly usage and savings of DQL agent for different households each using 3kWh storage in (b). Subfigure (c) displays solar energy generation patterns for June 2nd (Blue) and January 1st (Orange) generated by our simulation for a 1 kW PV. Subfigure (d) compares community demand without (Blue) and with (Orange) 0.5 kWh storage for all agents.

to case 3, mainly due to Agent 2's influence. This setup underscores that individual agents' savings hinge on the entire system's behaviour, and even households without storage or PV benefit when neighbours adopt these technologies.

7 Conclusions

In summary, our agent-controlled energy storage system benefits both consumers and suppliers, addressing the challenges of variable tariffs and contributing to SG development. Notably, all agents, even those initially disadvantaged, benefit thus fostering social acceptance of SG. Furthermore, the system smoothens demand curves, minimizing grid load fluctuations.

Key findings include: higher consumption correlates with smaller savings; increased storage capacity leads to greater bill reduction, plateauing around 4kWh; and significant savings result from combining the system with PV panels. Single-agent simulations favour immediate self-use, surplus storage, and excess energy sales, while MAS simulations prefer direct storage and surplus sale. The DQL agent maintains a full storage state most of the time, extending the battery lifespan. Octopus Agile price-demand patterns resemble a logarithmic function, with exponential pricing computations yielding optimal savings for MAS. Future work may involve refining the simulation to incorporate real-world dynamics, exploring larger agent populations, and adapting to current consumption patterns.

References

1. Ali, S.S., Choi, B.J.: State-of-the-art artificial intelligence techniques for distributed smart grids: A review. Electronics (2020)
2. Asare-Bediako, Kling, Ribeiro: Day-ahead residential load forecasting with artificial neural networks using smart meter data. In: IEEE Grenoble Conference (2013)
3. Atef, S., Eltawil, A.: A comparative study using deep learning and support vector regression for electricity price forecasting in smart grids. In: ICIEA (2019)
4. Barbour, E., Parra, D., Awwad, Z., González, M.C.: Community energy storage: A smart choice for the smart grid? Applied Energy (2018)
5. Bayram, et al.: A survey on energy trading in smart grid. In: GlobalSIP (2014)
6. Bellman, R.: Dynamic Programming. Dover Publications (1957)
7. Brunton, S.L., Kutz, J.N.: Data Driven Science and Engineering Machine Learning, Dynamical Systems, and Control. Cambridge University Press (2021)
8. Deng, Yang, Chow, Chen: A survey on demand response in smart grids: Mathematical models and approaches. IEEE Transactions on Industrial Informatics (2015)
9. Department for Business, Energy & Industrial Strategy: "Energy Trends: UK renewables" (January 2023)
10. Eck, Fusco, Gormally, Purcell, Tirupathi: AI modelling and time-series forecasting systems for trading energy flexibility in distribution grids. In: ACM e-Energy (2019)
11. Ghasemi et al.: A multi-agent deep reinforcement learning approach for a distributed energy marketplace in smart grids. In: SmartGridComm (2020)
12. Guy Lipman: Forecasting UK electricity prices. Medium blog post (July 2020)
13. Hassaballah, Awad: Deep learning in computer vision: principles and applications. CRC Press (2020)
14. Khan, Mahmood, Safdar, Khan, Khan: Load forecasting, dynamic pricing and dsm in smart grid: A review. Renewable and Sustainable Energy Reviews (2016)
15. Khoshjahan, Soleimani, Kezunovic: Optimal participation of pev charging stations integrated with smart buildings in the wholesale energy and reserve markets. In: IEEE Power Energy Society Innovative Smart Grid Technologies (2020)
16. Lin, L.J.: Reinforcement Learning for Robots Using Neural Networks. Ph.D. thesis, Carnegie Mellon University (1992)
17. Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M.: Playing atari with deep reinforcement learning (2013)
18. Octopus Energy Group: Octopus agile tariff (November 2022)
19. Ofgem: "Feed-in Tariffs (FIT)" (March 2023)
20. Qiu, Ye et al.: Scalable coordinated management of peer-to-peer energy trading: A multi-cluster deep reinforcement learning approach. Applied Energy (2021)
21. Roesch, Linder, Zimmermann, Rudolf, Hohmann, Reinhart: Smart grid for industry using multi-agent reinforcement learning. Applied Sciences (2020)
22. Schofield, Carmichael, Tindemans, et al.: Low carbon london project: Data from the dynamic time-of-use electricity pricing trial. uK Data Service, SN (2015)
23. Singh, et al.: Distributed multi-agent system-based load frequency control for multi-area power system in smart grid. Transactions on Industrial Electronics 2017
24. Song, Jiang, Tu, Du, Neyshabur: Observational overfitting in reinforcement learning. In: International Conference on Learning Representations (2020)
25. Voice, T., Vytelingum, P., Ramchurn, S., Rogers, A., Jennings, N.: Decentralised control of micro-storage in the smart grid. AAAI (2011)
26. Wang, Gu, Li, Bale, Sun: Active demand response using shared energy storage for household energy management. IEEE Transactions on Smart Grid (2013)
27. Yarats, Zhang, Kostrikov, Amos, Pineau, Fergus: Improving sample efficiency in model-free reinforcement learning from images. In: CoRR (2019)

A Experimental Setup details

Algorithm 1 DQL reward policy

if action == hold **then**
  if Price (p/kWh) $\geq$ 10.71 & storage not empty & Value of energy in storage (p/kWh) $\geq$ Price (p/kWh) **then**
    Reward = 1
  else
    Reward = 0.001
  end if
else if action == buy **then**
  if Storage = empty & Price (p/kWh) $<$ 10.71 **then**
    Reward = 1
  else if Price per kWh $\geq$ 10.71 **then**
    Reward = 0
  else
    Reward = 0.3
  end if
else if action == use **then**
  if Price (p/kWh) $\geq$ Value of energy in storage (p/kWh) & Storage not empty & Import price (p/kWh) $\geq$ Export price (p/kWh) **then**
    Reward = 1
  else
    Reward = -0.3
  end if
else if action == sell **then**
  if Export price (p/kWh) $\geq$ Value of energy in storage (p/kWh) & Export price (p/kWh) $\geq$ Import price (p/kWh) & Storage not empty **then**
    Reward = 1
  else
    Reward = -0.3
  end if
end if

B Simulation of PV energy generation details

Furthermore, real-world data has brief periods of low energy production due to cloud cover. To simulate that, the amount of energy produced in an interval chosen with a 15% probability is multiplied by 0.1 if the previous interval was not cloudy. Otherwise, the probability increases to 60%, creating longer periods of overcast weather.

The average monthly energy production is estimated using the widely available website https://www.renewables.ninja/. These values are then divided by 30 to estimate the daily energy production for each month.

C Explanation of inherent regularization in MAS with prosumers

The solar energy generation data is included in the MAS simulation with a logarithmic price scheme to model some households as prosumers. The selling

price is established at 80% of the import price, unless the import price is negative, in which case the selling price is set to 0. This mirrors the import-export price relationship seen in the Octopus Agile tariff. This simplified relationship effectively simulates real-world scenarios due to inherent regularization. In our simulation, high export prices would require high import prices, which usually occur during high community demand. However, in such cases, agents seldom have surplus energy to sell, making the impact of high export prices negligible in the simulation.

D Results details

Table 4: Yearly savings resulting from using Octopus Agile tariff versus a constant tariff of 14.228pence/kWh. Columns are independent (maximum values in £ and % may not come from the same test dataset).

Metrics	Savings in £	Saving as % of total bill
Median	42.78	12.34
Mean	60.78	13.65
Standard Deviation	59.77	9
Maximum value	263.61	35.69
Minimum value	-14.02	-3

Table 5: Difference of profits per year between variable and flat (3.54pence/kWh) export tariffs on solar energy generation simulations with different powers used for single agent investigation. The columns are independent.

	1kW installation		4kW installation		6kW installation	
Metrics	Profit in £	Profit in %	Profit in £	Profit in %	Profit in £	Profit in %
Median	42.1	139.61	171.63	138.96	257.54	139.16
Mean	35.75	118.63	147.74	118.61	221.21	118.47
Standard Deviation	13.37	44.05	55.21	44.23	81.96	43.7
Maximum value	52.52	175.21	216.35	173.35	324.85	175.99
Minimum value	9.54	30.99	39.98	31.64	57.56	31.05

Table 6: Savings showed as a percentage ratio between costs with and without storage for agents in columns 1-3. Columns 4-7 display the percentage change in savings between cases when all, and not all agents have storage. All agents use 0.5 kWh storage in a shared market with a linear price-demand function.

Agents with battery	Savings (%)			Savings Change (%)		
	Agent 0	Agent 1	Agent 2	Agent 0	Agent 1	Agent 2
0, 1 and 2	3.16 ± 0.45	2.73 ± 0.38	4.75 ± 0.29	-	-	-
1 and 2	2.88 ± 0.27	2.68 ± 0.39	4.66 ± 0.20	- 8.86	- 1.83	- 1.89
0 and 2	1.73 ± 0.54	1.27 ± 0.08	4.72 ± 0.10	- 45.25	- 53.48	- 0.63
0 and 1	2.27 ± 0.28	2.70 ± 0.74	2.08 ± 0.51	- 28.16	- 1.10	- 56.21
2	1.42 ± 0.36	1.21 ± 0.07	4.63 ± 0.17	- 44.94	- 55.68	- 2.53
1	1.98 ± 0.19	2.66 ± 0.74	1.95 ± 0.44	- 37.34	- 2.56	- 58.95
0	0.32 ± 0.20	0.06 ± 0.01	0.13 ± 0.07	- 89.87	- 97.80	- 97.26

E Discussion details

Recall a household initially incurring a £14.02 loss upon switching to a variable tariff. However, employing the 3-action DQL agent results in substantial savings: £21.24 (4.41%) for 0.5 kWh, £62.35 (12.93%) for 1.5 kWh, and £96.3 (19.98%) for 3 kWh storage. Compared to flat tariff costs, this equates to savings of 1.69%, 11.3%, and 19.25% for the respective storage capacities. This indicates that even households potentially affected by variable tariffs in the SG can benefit from adopting the proposed storage system.

When analyzing agent behaviour, it's crucial to distinguish between successful and unsuccessful actions. An action is deemed unsuccessful when the agent tries to purchase energy with full storage or utilize electricity with an empty battery. This unsuccessful action is comparable to waiting, as it doesn't affect the environment. For detailed statistics on the frequency of action selection in the test dataset, please consult Table 7.

Table 7: Frequency of a given action occurrence for different agents in % in the test dataset. An action is unsuccessful when the agent attempts to buy energy with full storage or use electricity with an empty battery. Successful buy or use actions are when energy is actually bought or used.

Agent number	Wait action	Successful buy action	Successful use action	Unsucc. buy action	Unsucc. use action
0	46.16	0.09	0.13	53.63	0
1	41.05	3.57	5.44	39.91	10
2	16.35	2.12	2.89	78.65	0

A detailed examination of agents' behaviour reveals no clear signs of cooperation or competition. Agents seem to recognize the influence of others on prices, with instances of more than one agent successfully purchasing or using energy being infrequent, constituting only 1.21% and 1.77% of the total amount of actions, respectively. These cases, where significant differences between initial and final prices occur, could be interpreted as competitive behaviours. Conversely, scenarios where one agent successfully buys, another successfully uses, and a third waits result in minimal price changes, suggesting potential cooperative behaviour. However, such cases are rare, amounting to only 0.05% of total episodes. These findings indicate that selfish agents do not establish implicit relationships with each other.

The difference between the best strategies for generated energy utilization in single-agent and MAS simulations can be attributed to the price-demand relationship in MAS. Storing generated energy and using it only during favourable moments, as observed in the MAS agents' approach, proves more effective in maximizing savings. In MAS, optimal savings are achieved when electricity generation moderately exceeds individual agent demand, highlighting the importance of maintaining export price viability without driving it down. This assumes other agents don't adopt the same strategy, which could impact profitability.