Exploring market power using deep reinforcement learning for intelligent bidding strategies

Alexander J. M. Kell, Matthew Forshaw, A. Stephen McGough
School of Computing
Newcastle University
Newcastle upon Tyne, U.K.
{a.kell2, matthew.forshaw, stephen.mcgough}@newcastle.ac.uk

***Abstract*—Abstract goes here.**

***Index Terms*—deep reinforcement learning, bidding strategy, multi-agent system, electricity markets**

I. Introduction

Under perfectly competitive electricity markets, generator companies (GenCos) tend to bid their short-run marginal costs (SRMC) when bidding into the day-ahead electricity market. SRMC is the cost to produce a single MWh of electricity and excludes capital costs. However, electricity markets are often oligopolistic, where a small subset of GenCos provide a majority of the capacity to the market. Under these conditions, it is possible that the assumption that GenCos are price-takers does not hold. That is, large GenCos artificially increase the price of electricity to gain increasing profit using their market power.

Reduced competition within electricity markets can lead to higher prices to the consumers, for no increased societal benefit. It is, therefore, within the interests of the consumer and that of government to maintain a competitive market. Low energy costs can enable innovation in other industries reliant on electricity, and in turn, make a more productive economy. Competition within electricity markets can be decreased through having multiple entities with a relatively small control over capacity in electricity markets.

In this paper, we explore the effect of total controlled capacity on electricity prices. Specifically, we use deep reinforcement learning (RL) to calculate a bidding strategy for GenCos in a day-ahead market. These GenCos are modelled as agents within the agent-based model, ElecSim [1], [2]. We use the UK electricity market instantiated in 2018 as a case study, similar to our work in [3]. That is, we model each GenCo with their respective power plants in the year 2018 to 2019. In total, we model 60 GenCos with 1085 power plants.

We use the deep deterministic policy gradient (DDPG) deep RL algorithm, which allows for a continuous action space [4]. Conventional RL methods require discretization of state or action spaces and therefore suffer from the curse of dimensionality [5]. As the number of discrete states and actions increases, the computational cost grows exponentially. However, too small a number of discrete states and actions will reduce the information available to the GenCos, leading to sub-optimal bidding strategies. Additionally, by using a continuous approach, we allow for GenCos to consider increasingly complex bidding strategies.

Other work considers a simplified model of an electricity market by modelling a small number of GenCos or plants [6], [7]. We, however, model each GenCo as per the UK electricity market with their respective power plants in a day-ahead market. Additionally, other work focuses on a bidding strategy to maximize profit for a GenCo. In our work, we focus on the impact of large GenCos, as well as collusion, between GenCos on total the electricity price.

Our approach does not require GenCos to formulate any knowledge of the information informing the market clearing algorithm or rival GenCo bidding strategies, unlike in game-theoretic approaches [8]. This enables a more realistic simulation where the strategy of rival GenCos are unknown.

In Section II we review the literature, and explore other approaches of RL in electricity markets. In Section III we introduce the agent-based model used and the DDPG algorithm. Section IV explores the methodology taken for our case study. We discuss and conclude our work in Sections VI and VII respectively.

II. Literature Review

Intelligent bidding strategies for day-ahead electricity markets can be divided into two broad categories: game-theoretic models and those based upon simulation and agent-based models (ABMs). ABMs allow for the simulation of heterogenous irrational actors with imperfect information. Additionally, ABMs allow for the modelling of learning and adaption within a dynamic environment [6]. Game-theoretic approaches may struggle in complex electricity markets where Nash equilibriums do not exist [8].

Kumar *et al.* propose a Shuffled Frog Leaping Algorithm (SFLA) [9] to find bidding strategies for GenCos in electricity markets. SFLA is a meta-heuristic that is based on the evolution of memes carried by active individuals, as well as a global exchange of information among the frog population. They test the effectiveness of the SFLA algorithm on an IEEE 30-bus system and a practical 75-bus Indian system. They find superior results when compared to particle swarm optimization and the genetic algorithm with respect to total profit and

EPSRC

convergence with CPU time. They assume that each GenCo bids a linear supply function, and they model the expectation of bids from rivals as a joint normal distribution. In contrast to their work, we do not require an estimation of the rivals bids.

Wang *et al.* propose an evolutionary imperfect information game approach to analyzing bidding strategies with price-elastic demand [8]. Their evolutionary approach allows for GenCos to adapt and update their beliefs about an opponents' bidding strategy during the simulation. They model a 2-bus system with 3 GenCos. Our work, however, models a simulation with 60 GenCos modelled across the entire UK.

The previously discussed approaches take a game-theoretic view. Next, we explore reinforcement learning approaches used to make intelligent bidding decisions in electricity markets. RL is a suitable method for analyzing the dynamic behaviour of complex systems with uncertainties. RL can therefore be used to identify optimal bidding strategies in energy markets [10].

Aliabadi *et al.* utilize an ABM and the Q-learning algorithm to study the impact of learning and risk aversion on GenCos in an oligopolistic electricity market with five GenCos [6]. They find that some level of risk aversion is beneficial, however excessive risk degrades profits by causing an intense price competition. Our paper focuses on the impact of the interaction of many GenCos within the UK electricity market. In addition, we extend the Q-learning algorithm to use the DDPG algorithm, which uses a continuous action.

Bertrand *et al.* use RL in a continuous intraday market. Specifically, they use the REINFORCE algorithm to optimize the choice of price thresholds. They demonstrate an ability to outperform the traditionally used method, the rolling intrinsic method, by increasing profit per day by 4.2%. The rolling intrinsic method accepts any trade, which gives a positive profit if the contracted quantity remains in the bounds of capacity. In our paper, we model a day-ahead market and use a continuous action for price bids, as opposed to a discrete threshold.

Ye *et al.* propose a novel deep RL based methodology which combines the DDPG algorithm with a prioritized experience replay (PER) strategy. The PER samples from previous experience, but samples from the "important" ones more often [11]. They use a day-ahead market as a case study with hourly resolution and show that they are able to achieve approximately 41%, 20% and 11% higher profit for the GenCo than the MPEC, Q-learning and DQN methods, respectively. In our paper, we inspect the effect on electricity price within the UK for increasing sized groups of GenCos.

In [12], Zhao *et al.* propose a modified RL method, known as the gradient descent continuous Actor-Critic (GDCAC) algorithm. This algorithm is used in a double-side day-ahead electricity market simulation. Where in this case, a double-side day-ahead market refers to GenCos selling their supply to distribution companies, retailers or large consumers. Their approach performs better in terms of participant's profit or social welfare compared with traditional RL methods.

III. Material

In this section we describe the RL methodology used for the intelligent bidding process as well as the simulation model used as the environment.

A. *Reinforcement Learning background*

Generally in RL, an agent interacts with an environment to maximize its cumulative reward. Generally, RL can be described as a Markov Decision Process (MDP). An MDP includes a state space $\mathcal{S}$, action space $\mathcal{A}$, a transition dynamics distribution $p(s_{t+1}|s_t, a_t)$ and a reward function, where $r : S \times \mathcal{A} \rightarrow \mathbb{R}$. At each time-step an agent receives an observation.

An agent's behaviour is defined by a policy, π. π maps states to a probability distribution over the actions $\pi : \mathcal{S} \rightarrow \mathcal{P}(\mathcal{A})$. The return from a state is defined as the sum of discounted future reward $R_t = \sum_{i=t}^{T} \gamma^{(i-t)} r(s_i, a_i)$. Where γ is a discounting factor $\gamma \in [0, 1]$. The return is dependent on the action chosen, which is dependent on the policy π. The goal in reinforcement learning is the learn a policy that maximizes the expected return from the start distribution $J = \mathbb{E}_{r_i, s_i \sim E, a_i \sim \pi}[R_1]$.

The expected return after taking an action a_t in state s_t after following policy π can be found by the action-value function. The action-value function is used in many reinforcement learning algorithms and is defined in Equation 1.

$$Q^{\pi}(s_t, a_t) = \mathbb{E}_{r_{i \geq t}, s_{i>t} \sim \mathcal{E}, a_{i>t} \sim \pi}[R_t | s_t, a_t] \quad (1)$$

B. *Q-Learning*

An optimal policy can be derived from the optimal Q-values $Q_*(s_t, a_t) = \max_\pi Q_\pi(s_t, a_t)$ by selecting the action corresponding to the highest Q-value in each state.

Many approaches in reinforcement learning use the recursive relationship known as the Bellman equation. The Bellman equation is defined in Equation 2.

$$Q^{\pi}(s_t, a_t) = \mathbb{E}_{r_t, s_{t+1} \sim E}[r(s_t, a_t) + \gamma \mathbb{E}_{a_{t+1} \sim \pi}[Q_\pi(s_{t+1}, \pi(s_{t+1}))]] \quad (2)$$

The Q-value can therefore be improved by bootstrapping. This is where the current value of the estimate of Q_π is used to improve its future estimate, using the known $r(s_t, a_t)$ value. This is the foundation of Q-learning [13], a form of *temporal difference* (TD) learning [14], where the update of the Q-value after taking action a_t in state s_t and observing reward r_t, which results in state s_{t+1} is:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \delta_t \quad (3)$$

$$\delta_t = r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t) \quad (4)$$

where $\alpha \in [0, 1]$ is the step size, δ_t represents the correction for the estimation of the Q-value function and $r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1})$ represents the target Q-value at time step t.

It can be proven that if if the Q-value for each state action pair is visited infinitely often, the learning rate $alpha$ decreases over time step t, then as $t \to \infty$, $Q(s,a)$ converges to the optimal $Q_*(s,a)$ for every state-action pair [13].

However, it is the case that Q-learning suffers from the curse of dimensionality. This is because in Q-learning stores the Q-value function in a look-up table. This therefore requires the action and state spaces to be discretized. As the number of discretized states and actions increases, the computational cost increases exponentially, making the problem intractable.

C. Deep Deterministic Gradient Policy

It is not straightforward to apply Q-learning to continuous action spaces. This is because in continuous spaces, finding the greedy policy requires an optimization of a_t at every time-step. Optimizing for a_t at every time-step would be too slow to be practical with large, unconstrained function approximators and nontrivial action spaces [4]. To solve this, an actor-critic approach based on the deterministic policy gradient (DPG) algorithm is used [15].

The DPG algorithm maintains a parameterized actor function $\mu(s|\theta^{\mu})$ which specifies the current policy by deterministically mapping states to a specific action. The critic $Q(s,a)$ is learned using the Bellman equation as in Q-learning. The actor is updated by applying the chain rule to the expected return from the start distribution J with respect to the actor parameters:

$$\nabla_{\theta^{\mu}} J \approx \mathbb{E}_{s_t \sim \rho^{\beta}}[\nabla_{\theta^{\mu}} Q(s,a|\theta^{Q})|_{s=s_t, a=\mu(s_t|\theta^{\mu})}] \quad (5)$$

$$= \mathbb{E}_{s_t \sim \rho^{\beta}}[\nabla_a Q(s,a|\theta^{Q})|_{s=s_t, a=\mu(s_t)} \nabla_{\theta_{\mu}} \mu(s|\theta^{\mu})|_{s=s_t}] \quad (6)$$

This is the policy gradient. The policy gradient is the gradient of he policy's performance.

Introducing non-linear function approximators, however, mean that convergence is no longer guaranteed. Although, these function approximators are required in order to learn and generalize on large state spaces. The Deep Deterministic Gradient Policy (DDPG) modifies the DPG algorithm by using neural network function approximators to learn large state and action spaces online.

A replay buffer is utilized in the DDPG algorithm to address the issue of ensuring that samples are independently and identically distributed. The replay buffer is a finite sized cache $\mathcal{R}$. This is where transitions are samples from the environment through the use of the exploration policy, and the tuple (s_t, a_t, r_t, s_{t+1}) is stored in the replay buffer. Old samples are discarded as the replay buffer becomes full. The actor and critic are trained by sampling from the replay buffer uniformly.

A copy is made of the actor and critic networks, $Q'(s,a|\theta^{Q'})$ and $\mu'(s|\theta^{\mu'})$ respectively. These are used fr calculating the target values. To ensure stability, the weights of these target networks are updated by slowly tracking the learned networks.

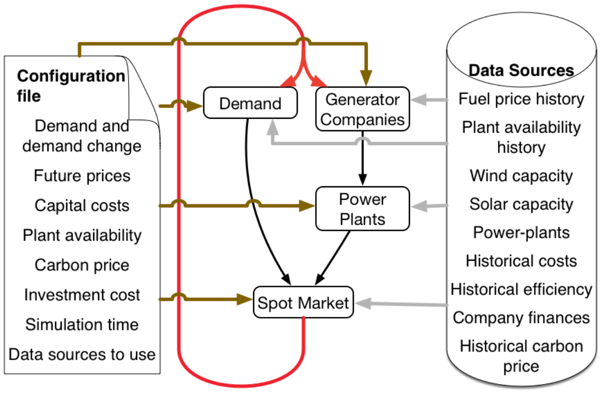

Fig. 1. System overview of ElecSim [1].

D. Simulation

For this work we utilized the long-term electricity market agent-based model, ElecSim [1], [16]. We utilized the model in a short term approach, however, by only iterating through a single year (2018), composed of eight representative days, each of 24 time-steps.

ElecSim is made up of six fundamental components: 1) power plant data; 2) scenario data; 3) the time-steps of the algorithm; 4) the power exchange; 5) the investment algorithm and 6) the generation companies (GenCos) as agents. For this paper we ignore the investment algorithm, due to investments happening only once a year, and not in the first year of operation.

ElecSim uses a subset of representative days of electricity demand, solar irradiance and wind speed to approximate a full year. Representative days, in this context, are a subset of days which can closely approximate an entire year's electricity demand and weather patterns. By using a subset, we are able to reduce the computational time to run for a year, whilst maintaining accuracy of modelling the real electricity market [2].

Figure 1 shows how the six fundamental components interact. The electricity demand is matched with the supply of the power plants, owned by the generator companies (GenCos). We have configured ElecSim to model the UK electricity market, using the configuration file. Specifically, we model the actual GenCos with their respective power plants that were in operation in 2018.

The market has a uniform pricing bidding mechanism. A uniform pricing mechanism is one where a single price is paid for all electrical capacity, irrespective of the bid. The bid that is paid is that of the highest accepted price. This incentivises GenCos to bid their SRMC is the price that it takes to generate a single MWh of electricity, excluding capital costs.

In this work we modify the bidding strategy of a subset of the GenCos. They bid based upon the policy of the DDPG RL algorithm, as opposed to their SRMC. This is to explore whether large GenCos, or a group of GenCos can manipulate the price of the electricity market through market power. The

GenCo Groups	Capacity	Num. of Plants
Orsted	2738.7	11
Drax Power Ltd	4035.0	3
Scottish power	4471.5	49
Uniper UK Limited	6605.0	9
SSE	8390.7	130
RWE Generation SE	8664.0	11
EDF Energy	14763.0	14
{EDF Energy, RWE Generation SE}	23427.0	25
{EDF Energy, RWE Generation SE, SSE}	31817.7	155
{EDF Energy, RWE Generation SE, SSE, Uniper UK Ltd}	38422.7	164
{EDF Energy, RWE Generation SE, SSE, Uniper UK Ltd, Scottish Power}	42894.2	213
{EDF Energy, RWE Generation SE, SSE, Uniper UK Ltd, Scottish Power, Drax Power Ltd}	46929.2	216

TABLE I

GROUPS OF GENCOS THAT USED BIDDING STRATEGIES, NUMBER OF PLANTS AND TOTAL ELECTRICITY GENERTING CAPACITY OWNED.

remaining GenCos, which fall outside of this group, maintain a bidding strategy based upon their SRMC.

For the purpose of this work we do not consider flow constraints within the electricity mix. This is because we model the entire UK with over 1000+ generators, and many nodes and buses. This would make the optimization problem intractable for the purpose of our simulation, especially when considering the many episodes required for training. It takes $\sim$10 seconds to run a single year in the simulation, or episode with our current setup.

IV. METHODOLOGY

To parameterize the simulation we use data from the United Kingdom in 2018. This included 1085 electricity generators and power plants with their respective GenCos. We changed the bidding strategy of select GenCos, as well as groups of GenCos, from a bid based upon the SRMC to a policy defined by the DDPG RL algorithm.

Through this we hoped to observe the ability for RL to find the point at which market power artificially inflates electricity prices. To achieve this we chose the six largest GenCos in the UK, as well as a smaller GenCo as a control. Table I displays the groups of GenCos with their respective capacity and number of plants.

For the reinforcement learning problem we have the following tuple: (s_t, a_t, r_t, s_{t+1}), where (s_t, s_{t+1}) is the state at time t and $t+1$ respectively, a_t is the action at time t and r_t is the reward at time t. For our problem the state space is given by the tuple shown in Equation 7:

$$s_t = (H_i, D_i, p_{C02}, p_{gas}, p_{coal}, p_c) \tag{7}$$

where H_i is the segment hour to bid into at timestep i, D_t is the demand of the segment hour at timestep t, p_{gas} is the price of gas, p_{C02} is the carbon tax price, and p_c is the clearing price. We set the reward, r_t to be the average electricity price of that time-step, p_{avg}.

For the action space, a_t, we modelled two scenarios. Where there was a price cap of 150/MWh and 600/MWh. We chose 150/MWh as a reasonable price cap that may be introduced by a Government, which allowed for variations in demand within different demand segments. The 600/MWh was chosen to simulate an unbounded price cap. This enabled us to see the price that an equilibrium is reached within a market with agents with market power.

For this work, we assume that the action space a_t only bids price, and not how much capacity to bid on the market. We assume this to reduce the dimensionality of a_t, and simplify the training process.

In this work, we assume that the GenCo Groups have no information about the generation capacity, marginal cost, bid prices or profits of other GenCos [6]. They learn through experience the maximum profit that can be made through experience within a particular market.

V. RESULTS

- Show time-steps vs. reward for both scenarios
- Show the step change in reward after a certain amount of controlled capacity

VI. DISCUSSION

- Make suggestions based upon optimal level of competition
- Importance of understanding market power and having a regulator otherwise prices can significantly increase

VII. CONCLUSION

- Future work (withold capacity)

VIII. ACKNOWLEDGMENT

This work was supported by the Engineering and Physical Sciences Research Council, Centre for Doctoral Training in Cloud Computing for Big Data [grant number EP/L015358/1].

REFERENCES

[1] A. Kell, M. Forshaw, and A. S. Mcgough, "ElecSim : Monte-Carlo Open-Source Agent-Based Model to Inform Policy for Long-Term Electricity Planning," *The Tenth ACM International Conference on Future Energy Systems (ACM e-Energy)*, pp. 556–565, 2019.

[2] A. J. M. Kell, M. Forshaw, and A. S. McGough, "Long-Term Electricity Market Agent Based Model Validation using Genetic Algorithm based Optimization," *The Eleventh ACM International Conference on Future Energy Systems (e-Energy'20)*, 2020.

[3] A. Kell, M. Forshaw, and A. S. McGough, "Modelling carbon tax in the UK electricity market using an agent-based model," *e-Energy 2019 - Proceedings of the 10th ACM International Conference on Future Energy Systems*, no. Ldc, pp. 425–427, 2019.

[4] J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous learning control with deep reinforcement," *International Conference on Learning Representations (ICLR)*, 2016.

[5] Y. Ye, D. Qiu, M. Sun, D. Papadaskalopoulos, and G. Strbac, "Deep Reinforcement Learning for Strategic Bidding in Electricity Markets," *IEEE Transactions on Smart Grid*, vol. 11, no. 2, pp. 1343–1355, 2020.

[6] D. Esmaeili Aliabadi, M. Kaya, and G. Sahin, "Competition, risk and learning in electricity markets: An agent-based simulation study," *Applied Energy*, vol. 195, pp. 1000–1011, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.apenergy.2017.03.121

[7] A. C. Tellidou and A. G. Bakirtzis, "Agent-based analysis of monopoly power in electricity markets," *2007 International Conference on Intelligent Systems Applications to Power Systems, ISAP*, 2007.

[8] J. Wang, Z. Zhou, and A. Botterud, "An evolutionary game approach to analyzing bidding strategies in electricity markets with elastic demand," *Energy*, vol. 36, no. 5, pp. 3459–3467, 2011. [Online]. Available: http://dx.doi.org/10.1016/j.energy.2011.03.050

[9] J. Vijaya Kumar and D. M. Kumar, "Generation bidding strategy in a pool based electricity market using Shuffled Frog Leaping Algorithm," *Applied Soft Computing Journal*, vol. 21, pp. 407–414, 2014. [Online]. Available: http://dx.doi.org/10.1016/j.asoc.2014.03.027

[10] T. Yang, L. Zhao, W. Li, and A. Y. Zomaya, "Reinforcement learning in sustainable energy and electric systems: a survey," *Annual Reviews in Control*, vol. 49, pp. 145–163, 2020.

[11] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, "Prioritized experience replay," *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*, pp. 1–21, 2016.

[12] H. Zhao, Y. Wang, S. Guo, M. Zhao, and C. Zhang, "Application of a Gradient Descent Continuous Actor-Critic Algorithm for Double-Side Day-Ahead Electricity Market Modeling," *Energies*, vol. 9, no. 9, p. 725, 2016.

[13] C. J. C. H. Watkins and P. Dayan, "Q-Learning," *Machine Learning*, vol. 292, pp. 179–184, 1992.

[14] R. S. Sutton and A. G. Barto, "An introduction to reinforcement learning," *The MIT Press*, 2015.

[15] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," *31st International Conference on Machine Learning, ICML 2014*, vol. 1, pp. 605–619, 2014.

[16] A. J. M. Kell, M. Forshaw, and A. S. McGough, "Long-term electricity market agent based model validation using genetic algorithm based optimization," pp. 1–13, 2020. [Online]. Available: http://arxiv.org/abs/2005.10346