

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—Decentralized electricity markets are often dominated by a small set of generator companies who control the majority of the capacity. In this paper, we explore the effect of the total controlled electricity capacity by any single or group of generator companies on the average electricity price. We demonstrate this through ElecSim, a simulation of a country-wide energy market. We develop an agent, representing a power generator which uses a deep deterministic policy reinforcement learning algorithm to bid in a uniform pricing electricity market strategically. A uniform pricing market is one where all players are paid the highest accepted price. ElecSim is parameterized to the United Kingdom for the year 2018. This work can help inform policy on how to best regulate a market to ensure that the price of electricity remains competitive.

We find that capacity has an impact on the average electricity price in a single year. If any single generator company, or a collaborating group of generator companies, control more than $\sim 11\%$ of generation capacity, prices begin to increase by $\sim 25\%$. The value of $\sim 25\%$ and $\sim 11\%$ may vary between market structures and countries. For instance, different load profiles may favour a particular type of generator or a different distribution of generation capacity. Once the capacity controlled by a generator company is higher than $\sim 35\%$ of the total capacity, prices increase exponentially. The use of a market cap of approximately double the average market price has the effect of significantly decreasing this effect and maintaining a competitive market.

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

I. INTRODUCTION

In this paper, we use a monte-carlo simulation to model a decentralized electricity market. Decentralized electricity markets have a number of different market structures: day-ahead, balancing market and reserve markets. The day-ahead market settles payments for electricity for the following day. The balancing market is designed to regulate the frequency of the market through matching supply and demand. Whereas, reserve markets provide capacity in case there are fluctuations in supply and demand, and electricity demand can't be matched with supply. Our model simulates a day-ahead market over a single year time period to explore the effects of market power on electricity prices.

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 the 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 an increased profit using their market power. If they were price-takers, they would have to accept the competitive price set by the market.

Reduced competition within electricity markets can lead to higher prices to consumers, with no societal benefit. It is, therefore, within the interests of the consumer and that of government to maintain a competitive market. Low energy costs enable innovation in other industries reliant on electricity, and in turn, make for a more productive economy.

In this paper, we explore the effect of total control over capacity on electricity prices. Specifically, we model different sizes of GenCos and groups of colluding GenCos, to bid strategically to maximize their profit using a reinforcement learning algorithm. This is in contrast to the strategy of bidding using the SRMC of their respective power plants, which would occur under perfect market conditions. To model this 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 on 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. It is possible, however, to generalise this approach and model to any other decentralized electricity market.

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 works have considered 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. In addition, further work focuses on a bidding strategy to maximize profit for a GenCo. However, in our work, we focus on the impact that large GenCos, or colluding groups of GenCos, can have on electricity price.

Our approach does not require GenCos to formulate any knowledge of the data which informs 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 uses 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. The results are presented in Section V. 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 simulation. Agent-based models (ABMs) allow for the simulation of heterogeneous irrational actors with imperfect information. Additionally, ABMs allow for learning and adaption within a dynamic environment [6]. Game-theoretic approaches may struggle in complex electricity markets where Nash equilibria do not exist [8].

A. Game-theoretic approaches

Here, we explore game-theoretic approaches. 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. A bus in a power system is a vertical line which several components are connected in a power system. For example, generators, loads, and feeders can all be connected to a bus. Kumar *et al.* find superior results when compared to particle swarm optimization and the genetic algorithm with respect to total profit and convergence with CPU time. They assume that each GenCo bids a linear supply function, and 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 three GenCos. Our work, however, models a simulation with 60 GenCos across the entire UK, which would require a 28-bus system model [10].

B. Reinforcement learning to model intelligence

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 [11]. Simulations are often used to provide an environment for the reinforcement learning algorithm. In the following papers simulations are used as the environment.

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 space.

Bertrand *et al.* use RL in an intraday market [12]. Specifically, they use the REINFORCE algorithm to optimize the choice of price thresholds. The REINFORCE algorithm is a gradient-based method. 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.

Ye *et al.* propose a novel deep RL based methodology which combines the DDPG algorithm with a prioritized experience replay (PER) strategy [5]. The PER samples from previous experience, but samples from the "important" ones more often [13]. The PER is a modification of the often used experience buffer, which is a buffer which stores previous transitions and samples uniformly. This helps to reduce the correlations between recent experiences. They use a day-ahead market 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 instead look at how to prevent GenCos (or sets of colluding GenCos) from forcing higher prices above market rates.

Zhao *et al.* propose a modified RL method, known as the gradient descent continuous Actor-Critic (GDCAC) algorithm [14]. This algorithm is used in a double-sided 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 table-based RL methods, such as Q-Learning. Our work also looks at improving on table-based methods by using function approximators.

III. METHODOLOGY

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

In RL an agent interacts with an environment to maximize its cumulative reward. 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 $r : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$. At each time step an agent receives an observation of the current state which is used to modify the agent's behaviour.

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 to 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.

$$Q^\pi(s_t, a_t) = \mathbb{E}_{r_i \geq t, s_i > t \sim \mathcal{E}, a_i > t \sim \pi}[R_t | s_t, a_t]. \quad (1)$$

The action-value function defines the expected reward at time t , given a state s_t and action a_t when under the policy π .

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, as defined in Equation 2:

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

The Bellman equation is equal to the action which maximizes the reward plus the discount factor multiplied by the next state's value, by taking the action after following the policy in state s_{t+1} or $\pi(s_{t+1})$.

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 [15], a form of *temporal difference* (TD) learning [16], 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:

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

where,

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

$\alpha \in [0, 1]$ 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 has been proven that if the Q -value for each state action pair is visited infinitely often, the learning rate α decreases over time step t , then as $t \rightarrow \infty$, $Q(s, a)$ converges to the optimal $Q_*(s, a)$ for every state-action pair [15].

However, it is often the case that Q-learning suffers from the curse of dimensionality. This is because 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. Many problems are naturally discretized which are well suited to a Q-learning approach, however this is not always the case.

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 [?]. To solve this, an actor-critic approach based on the deterministic policy gradient (DPG) algorithm is used [17].

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:

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

This is the policy gradient. The policy gradient is the gradient of the policy's performance. The policy gradient method optimizes the policy directly by updating the weights, θ , in such a way that an optimal policy is found within finite time. This is achieved by performing gradient ascent on the policy and its parameters π^θ .

Introducing non-linear function approximators, however, means 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} . Transitions are sampled from the environment through the use of the exploration policy, and the tuple (s_t, a_t, r_t, s_{t+1}) is stored within this replay buffer. \mathcal{R} discards older experiences as the replay buffer becomes full. The actor and critic are trained by sampling from \mathcal{R} 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 for calculating the target values. To ensure stability, the weights of these target networks are updated by slowly tracking the learned networks. Pseudo-code of the DDPG algorithm is presented in Algorithm 1.

Algorithm 1 DDPG Algorithm [4]

```

1: Initialize critic network  $Q(s, a|\theta^Q)$  and actor  $\mu(s|\theta^{\mu})$  with random
   weights  $\theta^Q$  and  $\theta^{\mu}$ 
2: Initialize target network  $Q'$  and  $\mu'$  with weights  $\theta^{Q'} \leftarrow \theta^Q, \theta^{\mu'} \leftarrow \theta^{\mu}$ 
3: Initialize replay buffer  $R$ 
4: for episode=1, M do
5:   Initialize a random process  $\mathcal{N}$  for action exploration
6:   Receive initial observation state  $s_1$ 
7:   for t=1, T do
8:     Select action  $a_t = \mu(s_t|\theta^{\mu}) + \mathcal{N}_t$  according to the policy and
        exploration noise,  $\mathcal{N}_t$ 
9:     Execute action  $a_t$  and observe reward  $r_t$  and new state  $s_{t+1}$ 
10:    Store transition  $(s_t, a_t, r_t, s_{t+1})$  in  $R$ 
11:    Sample a random minibatch of  $N$  transitions  $(s_i, a_i, r_i, s_{i+1})$ 
         from  $R$ 
12:    Set  $y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'})$ 
13:    Update critic by minimizing the loss:

$$L = \frac{1}{N} \sum_i (y_i - Q(s_i, a_i|\theta^Q))^2$$

14:    Update the actor policy using the sampled policy gradient:

$$\nabla_{\theta^{\mu}} J \approx \frac{1}{N} \sum_i \nabla_a Q(s, a|\theta^Q)|_{s=s_i, a=\mu(s_i)} \nabla_{\theta^{\mu}} \mu(s|\theta^{\mu})|_{s_i}$$

15:    Update the target networks:

$$\theta^{Q'} \leftarrow \tau \theta^Q + (1 - \tau) \theta^{Q'}$$


$$\theta^{\mu'} \leftarrow \tau \theta^{\mu} + (1 - \tau) \theta^{\mu'}$$

16:  end for
17: end for

```

D. Simulation

We utilized the long-term electricity market agent-based model, ElecSim [1], [2]. The model was run using a short term approach by only iterating through a single year (2018), composed of eight representative days, each of 24 time steps.

ElecSim is made up of six 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, whilst maintaining accuracy of modelling the real electricity market [2].

Figure 1 shows how the six 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

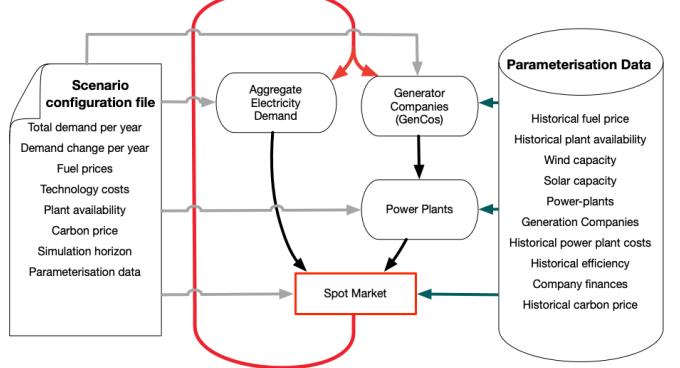


Fig. 1: System overview of ElecSim [1].

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 incentivizes GenCos to bid their SRMC. SRMC is the price that it takes to generate a single MWh of electricity, excluding capital costs.

In this work we explore whether large GenCos, or group of GenCos, can manipulate the price of the electricity market through virtue of their size. We achieve this by allowing a subset of GenCos to bid away from their SRMC and allow them to learn an optimal bidding strategy for maximizing their income. The GenCo agents adopt a DDPG RL algorithm to select their bids. This is to explore whether large GenCos, or a group of GenCos can manipulate the price of the electricity market through market power. The 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 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 ~ 125 seconds to run a single year in the simulation, or episode with our current setup. By increasing the simulation time further, we would make the compute time intractable due to the many episodes required for reinforcement learning to learn an effective policy. Additionally, we must train several reinforcement learning policies to account for each GenCo and market cap. Therefore a simulation that takes ~ 2 minutes to run, run 1000 times takes ~ 33 hours to run, multiplied by 12 different GenCos takes ~ 16 days, and with two market cap scenarios it would take ~ 1 month.

IV. EXPERIMENTAL SETUP

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. The data for this was taken from the BEIS DUKES dataset [18]. The electricity load data was modelled using data from [19]; offshore and onshore wind and solar irradiance data from [20]. It would be possible to adopt this approach to other decentralized markets in other countries.

By modelling bidding decisions as a RL algorithm, 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. Groups of GenCos are modelled as a single GenCo with a single RL strategy for the purpose of this paper. Table I displays the groups of GenCos, as well as individual 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 6:

$$s_t = (H_i, D_i, p_{CO2}, p_{gas}, p_{coal}, p_c) \quad (6)$$

where H_i is the segment hour to bid into at timestep i , D_i is the demand of the segment hour at timestep t , p_{gas} is the price of gas, p_{coal} the price of coal, p_{CO2} 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. Only these two values were chosen to reduce computational load. We chose £150/MWh as a reasonable price cap that may be introduced by a Government. This was roughly double the average accepted price in 2018, therefore allowed for higher prices in times of high demand or low supply. 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 the maximum profit that can be made through experience within a particular market. We assume this because in real-life GenCo groups have little information on their competitors sensitive bidding data. If they were to have perfect information on all their competitors they would be able to devise a perfect strategy which would always maximise their profit.

V. RESULTS

In this section, we detail the results of the RL algorithm, and the effect that capacity has on average electricity price within the UK. Our approach could be generalised to any other decentralised electricity market in other countries.

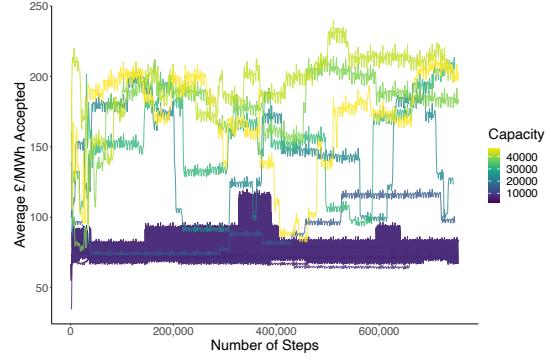


Fig. 2: Reward over time for different groups of GenCos, max bid = £600/MWh.

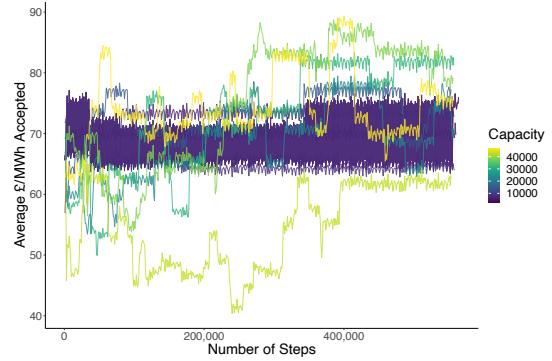


Fig. 3: Reward over time for different groups of GenCos, max bid = £150/MWh.

Figures 2 and 3 show the rewards over a number of time steps for the unbounded and bounded cases respectively. Figure 2 shows a clear difference between agents which use the DDPG RL strategy and have a large capacity (green and yellow) compared to those which have a smaller capacity (dark purple). The axis in Figure 2 are much larger than those of 3, highlighting the effect of market power on an unbounded market.

The average electricity price for a capacity below 30,000MW, or $\sim 35\%$ of total capacity, remains stable between £70/MWh and £100/MWh. This range may be due to the stochasticity in calculating the weights for the DDPG algorithm. The average electricity price does not change over the time steps or training. We, therefore, hypothesize that there is no market power as long as an individual GenCo owns below $\sim 35\%$ of total electrical capacity.

On the other hand, once the capacity of a GenCo or groups of GenCos is above 30,000MW, there is a significant increase in the average price for capacity. The average electricity price for capacity falls between the range of £170/MWh and £220/MWh.

Figure 4 displays the capacity controlled by the agents that use the RL strategy versus the average electricity price for the unbounded case. The color displays the number of steps. The step-change, as shown in Figure 2 can be seen clearly

GenCo Groups	Capacity (MW)	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 generating capacity.

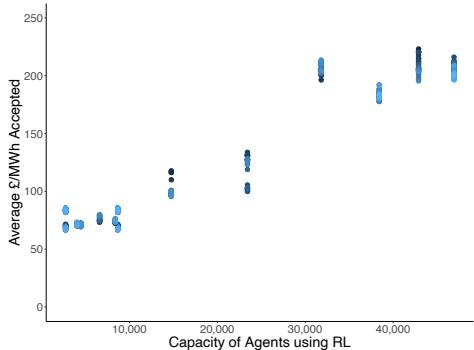


Fig. 4: Capacity of agents using RL vs. average electricity price accepted, for unbounded agents.

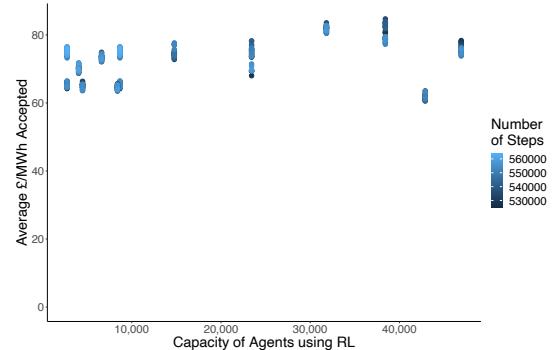


Fig. 5: Capacity of agents using RL vs. average electricity price accepted, for bounded agents.

here, with agents with a capacity larger than $\sim 25,000$ MW causing a step change in electricity price. Electricity prices seems to cluster below $\sim 10,000$ MW. However, after this point, the average electricity price begins to increase.

Figure 3 shows a cluster between $\sim \text{£}60/\text{MWh}$ and $\sim \text{£}80/\text{MWh}$ irrespective of the capacity of the agents. This is verified by Figure 5. This seems to suggest that setting a lower market cap reduces the ability for generators, irrespective of size, from influencing the electricity price.

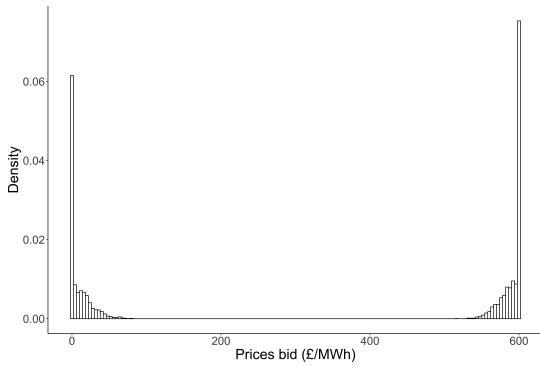
Figures 6 and 7 display the actual bids made at the end of training within the electricity market for all of their power plants. The number of bids made by each GenCo changes dependent on the number of plants that they own, with the Orsted GenCo only making eleven bids per segment, and the largest group making 216 bids per segment. Figure 6 displays the uncapped scenario (£600/MWh) and 7 displays the capped scenario (£150/MWh).

Figure 6a shows the bids made by the largest group of GenCos as shown in Table I. A bimodal distribution can be seen, where the group of GenCos tend to bid either the maximum or the minimum bid. We hypothesize that they bid the maximum amount as this ensures that the market price is artificially raised, and that they are able to utilize their market power. The minimum price is bid the rest of the time to ensure

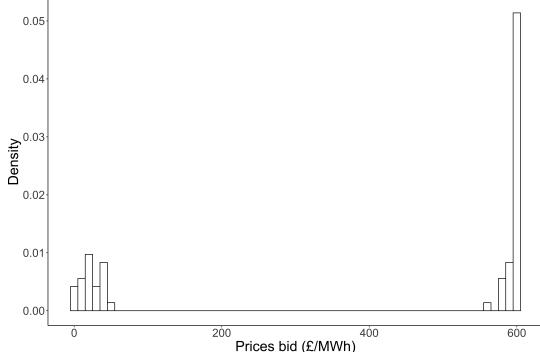
that generators bids are always accepted, regardless of whether the market price has been artificially raised or not in each respective clearing segment.

Figure 6b displays the bids of the small company with a market cap of £600/MWh. The small company also seems to have a bimodal distribution; however, bids the higher price more often. This may be due to the fact that it is able to influence the price at certain market segments, and the reward of the higher accepted reward outweighs the times in which it is not accepted on the market segments. Again, bidding low seems to be the strategy in which to take if the GenCo does not believe it will be able to influence the final price. As the market simulated is a uniform pricing market, having a £0 bid accepted does not mean that the GenCo will be paid £0. Rather, the GenCo will be paid the market clearing segment, which is set by the most expensive power plant accepted onto that market segment.

Figures 7a shows the bids made by the largest group of GenCos in each market segment. It seems to take a similar strategy to that of the large company in the uncapped scenario, as shown in Figure 6a. We believe, as similar in the uncapped scenario, that this is due to the market power that this group possesses. Being able to influence the market price enables the GenCo group to inflate the prices. It also takes a conservative



(a) Largest group of GenCos with a total controlled capacity of 46929.2MW with a market cap of £600/MWh.



(b) Smallest company with a total controlled capacity of 2738.7MW with a market cap of £600/MWh.

Fig. 6: Bids made by generator companies with a market cap of £600/MWh.

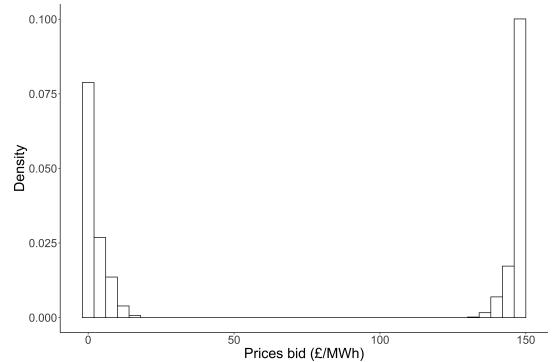
strategy to bid the minimum price allowed, to ensure that the bid is accepted regardless of price.

Figure 7b displays the strategy of the smallest company in the capped scenario. Here, it seems that the GenCo is unable to influence the price at all, and therefore bids £0/MWh for the majority of the time. This is similar to the expected strategy of GenCos, who tend to bid their short-run marginal cost to ensure that they do not miss out on potential profit. The short-run marginal cost can often change based upon fuel, carbon and generator type. However, for renewable energy it is near £0 and for fossil-fuel based plants it is near the cost of fuel and carbon at that point in time.

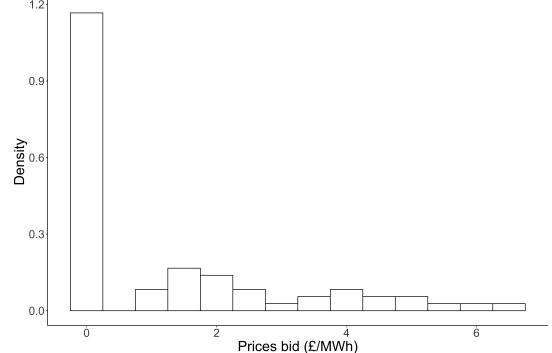
We ran a sensitivity analysis to observe the effects of the market cap on final average accepted bid price. The largest GenCo group used a strategy for this sensitivity analysis. Figure 8 displays the results. It seems that whilst the average accepted bid price increased with the capped bid level; there is a significant increase after a market cap of £190/MWh. This may be due to the case that the GenCos begin to outbid the SRMC bidding GenCos at this price point.

VI. DISCUSSION

Our results demonstrate the ability for GenCos to artificially increase the electricity price through market power in an



(a) Largest group of GenCos with a total controlled capacity of 46929.2MW with a market cap of £150/MWh.



(b) Smallest company with a total controlled capacity of 2738.7MW with a market cap of £150/MWh.

Fig. 7: Bids made by generator companies with a market cap of £150/MWh.

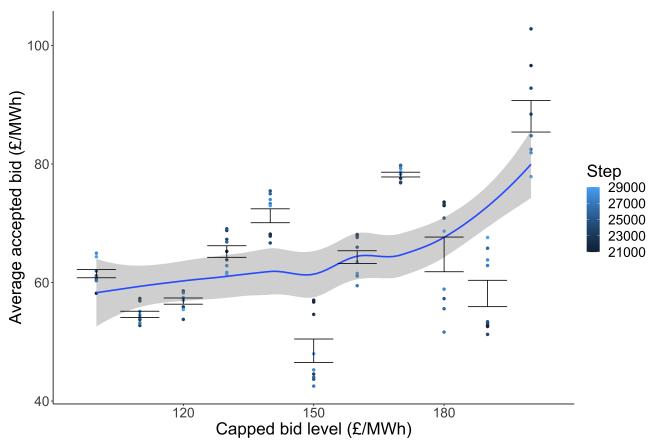


Fig. 8: Capacity of agents using RL vs. average electricity price accepted, for bounded agents.

uncapped market. Our results have shown that in an uncapped market, any single agent or groups of agents who make bids using the same strategy and information, should have less than $\sim 10,000$ MW. This defines the optimal capacity by any single GenCo to have a fair level of competition. After this, the electricity price begins to rise with the same outcome and welfare. It is also worth noting that when the market is capped, the average accepted price does not simply become the capped price.

However, if there is an electricity market with a few large or colluding players, it is possible to remove their advantage through the introduction of a price cap. Our results show that whilst average accepted bids increase with market cap level, the value does not increase significantly until £200 is reached.

This information and approach can help to inform government policy to ensure fair competition within electricity markets, as well as run the model for their own scenario. It is hypothesized that the findings in this paper are generalizable to other decentralized electricity markets in other geographies due to their similar market structures. Whilst the figures presented here may not be the same; we hypothesize that the region of interest will be similar.

VII. CONCLUSION

In this work, we used the deep deterministic policy gradient (DDPG) reinforcement learning method to make strategic bids within an electricity market. We utilized the agent-based model ElecSim to model the UK electricity market. We utilized the DDPG algorithm only for a certain subset of agents, from small individual generation companies (GenCos) to large groups of GenCos.

This enabled us to explore the ability for GenCos with a large capacity to artificially increase the price in the electricity market within the UK if they are in control of a sufficiently large generation capacity. Our results show that the optimum level of control of any single GenCo or groups of GenCo is below $\sim 10,000$ MW or $\sim 11\%$ of the total capacity. Above this, prices begin to increase with no real additional benefit to the consumer. After $\sim 25,000$ MW, or $\sim 35\%$ of the total capacity, the prices begin to increase substantially, to $\sim £200$, over triple the original cost without this market power. The introduction of a market cap of £150 reduces all market power and maintains electricity price at a reasonable level.

We found through a sensitivity analysis, that the average electricity price in the market over a year remains low with a price cap smaller than £190. However, after this level, the average electricity price begins to increase dramatically.

Our work has shown the ability for reinforcement learning to learn an optimal bidding strategy to maximize a GenCo's profit within an electricity market. The ability for GenCos to use their market power is also highlighted, and is dependent on electricity generation capacity of the respective GenCo.

In future work, we would like to enable GenCos to withhold the capacity on offer to the electricity market. This would enable further market power by reducing competition further. Additionally, we would like to assess the market power in

different countries with different market structures and total electricity supply.

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. Esmaeli Aliaabadi, 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.
- [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.
- [10] K. R. Bell and A. N. Tleis, "Test system requirements for modelling future power systems," *IEEE PES General Meeting, PES 2010*, no. August 2010, 2010.
- [11] 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.
- [12] G. Bertrand and A. Papavasiliou, "Reinforcement-Learning Based Threshold Policies for Continuous Intraday Electricity Market Trading," *IEEE Power and Energy Society General Meeting*, vol. 2019-Augus, 2019.
- [13] 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.
- [14] 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.
- [15] C. J. C. H. Watkins and P. Dayan, "Q-Learning," *Machine Learning*, vol. 292, pp. 179–184, 1992.
- [16] R. S. Sutton and A. G. Barto, "An introduction to reinforcement learning," *The MIT Press*, 2015.
- [17] 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.
- [18] D. for Business Energy and I. Strategy, "Power stations in the united kingdom, may 2019," *Digest of United Kingdom Energy Statistics (DUKES)*, 2019.
- [19] "Gb fuel type power generation production," 2020. [Online]. Available: <https://gridwatch.co.uk/>
- [20] S. Pfenninger and I. Staffell, "Long-term patterns of European PV output using 30 years of validated hourly reanalysis and satellite data," *Energy*, vol. 114, pp. 1251–1265, 2016.