Pereira *et al. Energy Informatics* 2022, **5**(Suppl 4):44
https://doi.org/10.1186/s42162-022-00235-2

Energy Informatics

RESEARCH Open Access

# Peer-to-peer energy trading optimization in energy communities using multi-agent deep reinforcement learning

Helder Pereira, Luis Gomes and Zita Vale*

*From* Energy Informatics.Academy Conference 2022 (EI.A 2022)
Vejle, Denmark. 24-25 August 2022

*Correspondence:
zav@isep.ipp.pt

Intelligent Systems Associated Laboratory (LASI), Research Group on Intelligent Engineering and Computing for Advanced Innovation and Development (GECAD), Polytechnic of Porto (P.PORTO), Rua Dr. António Bernardino de Almeida 431, 4200-072 Porto, Portugal

## Abstract

In the past decade, the global distribution of energy resources has expanded significantly. The increasing number of prosumers creates the prospect for a more decentralized and accessible energy market, where the peer-to-peer energy trading paradigm emerges. This paper proposes a methodology to optimize the participation in peer-to-peer markets based on the double-auction trading mechanism. This novel methodology is based on two reinforcement learning algorithms, used separately, to optimize the amount of energy to be transacted and the price to pay/charge for the purchase/sale of energy. The proposed methodology uses a competitive approach, and that is why all agents seek the best result for themselves, which in this case means reducing as much as possible the costs related to the purchase of energy, or if we are talking about sellers, maximizing profits. The proposed methodology was integrated into an agent-based ecosystem where there is a direct connection with agents, thus allowing application to real contexts in a more efficient way. To test the methodology, a case study was carried out in an energy community of 50 players, where each of the proposed models were used in 20 different players, and 10 were left without training. The players with training managed, over the course of a week, to save 44.65 EUR when compared to a week of peer-to-peer without training, a positive result, while the players who were left without training increased costs by 17.07 EUR.

**Keywords:** Energy communities, Local energy markets, Multi-agent deep reinforcement learning, Multi-agent systems, Peer-to-peer energy trading

## Introduction

Power systems are becoming increasingly distributed in terms of the management of the grid and the engagement of energy players. This allows the creation of smaller communities, such as microgrids and energy communities, usually composed of smart buildings (Mota et al. 2021), in which local energy management is carried out using local energy demand and renewable energy sources (RES). In this context,

energy communities are gaining importance, since they are seen as pillars of a successful energy transition (São et al. 2021), where they can alter the energy paradigm by empowering energy players, such as consumers and prosumers, therefore, contributing to energy and climate goals in what concerns meeting demand with renewable sources and reducing emissions (Reis et al. 2021).

There has been a substantial change in power systems, namely in energy consumption and generation, even more evident with the growth of distributed energy resources (DER) and the active engagement of prosumers, who are players that generate energy in addition to consuming it (Gržanić et al. 2022). The growth of active participant players reflects investments in the transition to RES. Nonetheless, it offers new challenges for the energy network, necessitating flexibility from energy players and efficient market mechanisms, which enable the correct operation of the smart grid. Besides that, the power and energy systems planning and operation become impaired or, at least, hampered, by the lower predictability and stability of RES generation (Chicco et al. 2021). With the shift in system structure, the energy market has to deal with a high number of small-scale renewable energy prosumers. Feed-in Tariff (FiT) system has been implemented as the most prevalent generation subsidy, in which prosumers get compensation for the energy exported to the grid at FiT supplied by the upstream utility firm (Chen and Liu 2021). This system enables prosumers to take advantage of the flexibility of their self-generated power and sell the surplus to the grid. Nevertheless, this payment may not consider the installation and operating cost of DER. On the other side, players who buy electricity from the grid are often faced with expensive Time-of-Use (ToU) tariffs supplied by the upstream utility provider (Venizelou et al. 2018).

In the smart grid paradigm, the concept of Transactive Energy (TE) is being highly addressed in order to balance the grid's consumption and generation. Several requirements, including two-way communication, integration of information and communication technology with the power grid, smart and remote supervision, and advanced and smart metering, are essential in this context (Wu et al. 2021). In reality, TE systems expand the present notions of wholesale transactive power systems into retail markets with end-users equipped with intelligent energy management systems to enable small energy consumers to participate actively in electricity markets (Abrishambaf et al. 2019). TE systems can also enable the administration of peer-to-peer (P2P) trading and local energy markets (LEM) in smart grids by employing devices with their own objectives and decision-making capabilities. A LEM can be used in a close community of prosumers and consumers which share a market platform for trading locally produced energy (Dudjak et al. 2021).

P2P energy trading comprises a large number of continuous data comprised of unpredictable and uncertain variables, such as renewable generation and load demand, making it difficult to make a decision using conventional optimization and learning techniques (Chen and Bu 2019). Thus, a commonly explored option is reinforcement learning, especially with the integration of deep learning and multi-agent techniques that give it a greater ability to approach optimal optimizations (Gronauer and Diepold 2021). A problem normally associated with this type of models is the difficulty in modelling the environment with which the agents will interact.

To tackle the problems mentioned here, the methodology proposed in this paper integrates the use of deep reinforcement learning to optimize the participation of players in LEMs and P2P markets. Two algorithms are used independently, which are integrated into an agent-based smart grid management ecosystem that allows agents to have access to innovative features that allow them to manage their resources in an intuitive, efficient and adaptable way. A case study was developed, using an energy community with players with real consumption and generation profiles, in order to test the proposed methodology. The results were positive and showed that this approach can create a competitive advantage for agents who train to participate in P2P, and also gives a greater ability to the community in general to transact more energy per period.

The paper is organized as follows, after the introduction, there is a section to present a general overview of the applications of Reinforcement Learning to local energy markets and P2P energy markets. Then, it is presented the proposed methodology, with a description of the Agent-based ecosystem for Smart Grid modelling (A4SG) and the training of both RL algorithms. The next section presents the case study, the used data and the description of the studied players, followed by a results section where is compared a P2P market week, before and after the RL training. Finally, the last section presents the main conclusions of the work.

## Reinforcement learning in peer-to-peer energy markets

Reinforcement learning (RL) is a type of trial-and-error learning in which an agent/learner interacts with its environment to learn the better action to take. Unlike other machine learning approaches, the agent is not advised on the right action to take. Instead, the agent explores the environment to maximize its future rewards (or, statistically, the sum of total expected rewards), generally in pursuit of a goal/objective represented numerically by a big reward (Recht 2019). The exponential growth of deep learning eventually extended to reinforcement learning, which had a beneficial effect on its potential applications. Deep reinforcement learning (DRL) combines the sensing ability of deep learning with the decision-making power of RL (Botvinick et al. 2019). Deep learning processes information about the target observation from the environment and delivers state information about the current environment. The RL algorithm then transfers the current state to the corresponding action and calculates predicted return values for each value (Arulkumaran et al. 2017). DRL handles standard RL challenges by performing complex tasks with less prior knowledge, as a result of its ability to learn abstraction levels from data (Arulkumaran et al. 2017).

With the increasing complexity of the problems addressed with the use of RL, namely in the smart grid, it was necessary to explore the cooperation and competition between RL agents, moving from a single-agent RL to a multi-agent RL (MARL) paradigm. In a MARL, there are two main approaches: (i) independent learning, which attempts to train a policy for each agent by mapping its private observations to an action, and (ii) cooperative learning, which aims to achieve a group goal by having each agent work on the issue as a whole or in subtasks. Dealing with multi-agent settings in competitive RL is not a straightforward problem, since agents are attempting to learn the best strategies to gain an edge over other agents in the same process, but with different configurations. Second, training independent policies typically does not

scale well to large numbers of agents, and the change in policies renders the environment dynamics non-stationary from the perspective of any particular agent, which may result in instability (Padakandla and Bhatnagar 2020). To overcome the non-stationarity issue, MARL methods have been employed to address this problem.

Prior applications of MARL in the field of power and energy systems are currently restricted but growing. Regarding the application of RL and MARL to LEMs and P2P energy trading, numerous learning and trading algorithms have been combined to increase energy consumers' involvement. One of the most explored algorithms is Q-Learning, and in Chiu et al. (2022) a multi-agent variant of this algorithm is presented to determine the optimal approach for energy market pricing negotiations. The most significant issue with Q-Learning, which prevents it from being used in an original manner in many smart grid situations, including the P2P market, is that it cannot deal with continuous observation and action spaces, and its adaptation, which involves the discretization of actions and observations, can result in the loss of valuable information, preventing optimal results. In Samende et al. (2022), it is proposed a multi-agent deep deterministic policy gradient (DDPG) algorithm using distribution network prices to incentivize an energy market in which RL incentives help to meet network limitations and choices that violate them.

Regarding the application of MARL to P2P energy trading, there are already several proposals with different methodologies. In Qiu et al. (2021a) it is proposed a multi-agent DDPG to automate P2P energy trading in double-auction markets. The proposed model provided a high level of scalability and also protects the privacy of prosumers/consumers by considering the market operator as a third-party service that provides agents with the market results. Also using multi-agent DDPG, but with a different approach, the work proposed in Qiu et al. (2021b) integrates the notion of parameter sharing to optimize the participation in P2P energy markets. This architecture allows all agents to share parameters (e.g., the weights of the actor and critic networks) of a single policy that is taught using the experiences of all agents (although each agent can obtain its unique observations). In Chen et al. (2022) it is proposed a model to optimize P2P energy trading and energy conversion policies of multi-energy microgrids in real-time. The proposed model uses twin delayed deep deterministic policy gradient algorithm (TD3) to improve the performance of the multi-agent actor-critic algorithm. The model considers P2P energy trading, energy conversion and multi-vector energies.

The proposed methodology in this paper, besides using multi-agent DDPG, uses TD3, proposed in Fujimoto et al. (2018). Even though DDPG can deliver good outcomes, it has limitations. As with many RL algorithms, DDPG training may be unstable and highly dependent on finding the optimal hyperparameters. This is because the algorithm consistently overestimates the Q values of the critic network. Over time, these errors may drive the agent to reach a local optimum or develop forgetfulness for prior experiences. TD3 solves this problem by concentrating on decreasing the overestimation bias. TD3 is an algorithm that tackles this problem by proposing three crucial techniques:

- Using a pair of critic networks: TD3 tends to underestimate Q values. This underestimating bias is not a concern because low values are not propagated through

the algorithm, unlike high values. This technique provides a more stable approximation, hence enhancing the algorithm's stability;

- Delayed updates of the actor: TD3 allows the definition of the delay periods to update the policy (and target networks) as an hyperparameter, thus updating it less often;
- Action noise regularization: when calculating the targets, clipped noise is added to the action. This makes that higher values are preferred for actions that are more robust.

Although there have been substantial advancements in the application of reinforcement learning to local energy markets and peer-to-peer energy trading, these models are trained for extremely narrow scenarios that do not allow for their implementation in real-time. The RL model proposed in this study was integrated and evaluated within an agent-based ecosystem designed to simulate smart grids, from which the model's execution data was obtained. Furthermore, the work proposed in this paper addresses the uncertainty that the forecast brings to participation in energy markets (Gomes et al. 2022), using the forecast error as a variable in the calculation of the amount of energy to be transacted.

### Double-auction market

The LEM implemented in this study is a P2P energy trading model based on the Double Auction (DA) (Friedman 2018). This is a particularly interesting negotiation model for integration with reinforcement learning models, because it motivates agents to look for different negotiation strategies when compared to their opponents. In DA traders are allowed to submit bids/offers at the start of an auction period, following which the auctioneer clears the market and publishes the results, i.e., trading prices and quantities.

More in detail, a DA market is comprised of a group of buyers and a group of sellers who declare the quantity of energy they wish to trade, in kWh, as well as the minimum/maximum price they are willing to receive/pay for the energy they want to sell/buy, in EUR/kWh. The auctioneer then creates a public order book in which the accepted bids and offers are published. In the order book, buy orders are ordered by decreasing submitted buy prices, whereas sell orders are organized by increasing submitted sale prices. The DA market involves multiple steps, specifically in matching sellers and buyers. In the first step, when an auction period begins, market players input their order information as well as a trading price and energy quantity. All orders are recorded in the order book, and then the matching algorithm iterates through the order book and tries to match each sell order with a buy order until the selling price is greater than the buying price or there are no more mismatched buyers or sellers. Finally, when two orders are matched, the algorithm determines the market clearing price using the mid-price approach (Friedman 2018), as

$$ClearingPrice = \frac{BidPrice^{b} + BidPrice^{s}}{2} \tag{1}$$

where $BidPrice^{b}$ is the price that the buyer $b$ offered in the market, and $BidPrice^{s}$ is the price that the seller $s$ is asking, both in EUR/kWh. The transaction quantity is equal to the minimum quantity between the two matched orders. The whole process of market

clearing is represented in Fig. 1. In the double auction trading mechanism, each match between sellers and buyers has a clearing price, thus demanding exploring numerous strategies to ensure not only that the required energy is bought or sold, but also that transactions are done at the best possible pricing. This type of trading mechanism was chosen based on previous results in the literature (Friedman 2018), namely in what regards the application of RL algorithms (Qiu et al. 2021a).

## Proposed methodology

The methodology proposed in this paper aims to serve as decision support for participants in local energy markets, regarding the amount of energy to be transacted, the price bided/offered for that transaction, and the use of flexibility to counter possible extra costs related to participation in P2P. In this methodology, the players representation agents can use reinforcement learning-based training to improve their participation in P2P markets using simulation environments. For this, two DRL algorithms will be used, i.e., multi-agent versions of DDPG and TD3, separately, allowing the choice of agents who will take advantage of them. In order to facilitate the use of this methodology in real contexts, it was integrated into the Agent-based ecosystem for Smart Grid modelling (A4SG), from which agents can request training, and use the resulting policies in their participation in real-time. The A4SG is a multi-agent system framework developed by the authors to digitalize the smart grid operation models. Therefore, A4SG was used to integrate the proposed methodology due to its ability to provide agents with useful mechanisms that facilitate their active participation in the smart grid, as described in the following subsection.

### A4SG

The A4SG, conceived and developed by the authors, which architecture for integrating the proposed methodology is depicted in Fig. 2, combines the concepts of multi-agent systems (MAS) and agent communities (ACOM) to produce an ecosystem in which multiple agent-based systems can coexist and interact. ACOMs are smaller groupings of agents that can represent aggregation entities, such as energy communities. The use of several groups of agents allows a distributed and intelligent decision-making process, with the integration of different services in the groups, considering their objectives.

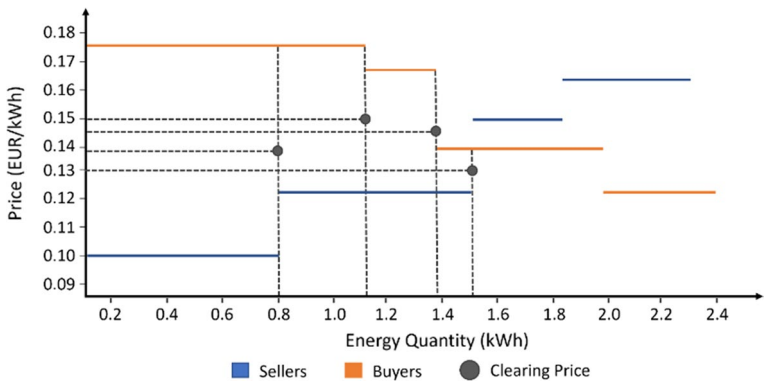


**Fig. 1** DA market clearing process

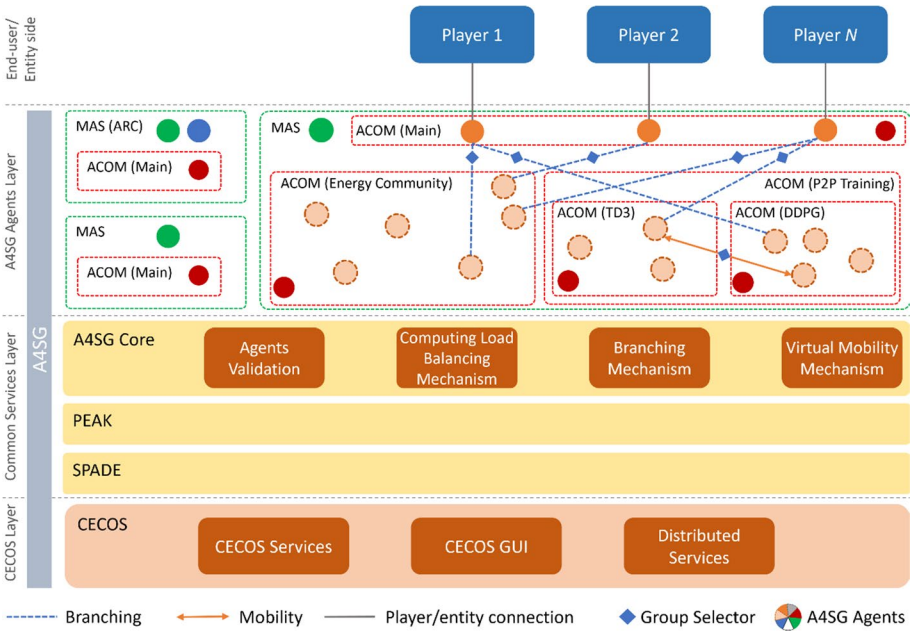


**Fig. 2** A4SG architecture to integrate the proposed methodology

Furthermore, this ecosystem takes advantage of two novel mechanisms, i.e., branching and mobility, to improve the agents' context and performance. The A4SG ecosystem is built on top of the Python-based Agent Communities Ecosystem (PEAK) (https://www.gecad.isep.ipp.pt/peak), and the Smart Python Agent Development Environment (SPADE) (Palanca et al. 2020), which enable the agents communication and distributed execution. Besides that, it uses as graphical interface the Citizen Energy Communities Operator System (CECOS) (Pereira et al. 2021), that enables the access to useful services, such as tariffs management, and demand response simulation.

As agents may have different objectives simultaneously, that involve the engagement in multiple ACOMs concurrently or even subscribe to various services, the branching mechanism was developed to offer this capability to the agents of the ecosystem. The branching of agents is the technique that permits the deployment of a new branch agent that acts as an extension of the representation agent to achieve a specific objective. There are two types of branch agents: the goal-oriented and the service-oriented agent. The goal-oriented tries to achieve an objective, which might be, for instance, the subscription of a service or the participation in an ACOM. On the other hand, the service-oriented agents provide services to other agents. In the context of this work, branching is important, since it allows an agent to have a representation in an energy community, and simultaneously deploy a goal-oriented agent to perform the RL training, only having the objective of returning the trained policy to the agent that later participates in the P2P market.

The agents' mobility in A4SG is divided in two types: the physical mobility, and the virtual mobility, supported by the Computation Load Balancing Mechanism and the Virtual Mobility Mechanism, respectively. The physical mobility within the ecosystem enables agents to move to a different physical location, for instance a different host, such as a computer or a server. From an individual point of view, the primary advantage is the

convenience that mobility may provide to the entity represented by the agent. From the ecosystem standpoint, the Computation Load Balancing Mechanism makes use of the physical mobility in A4SG to balance the available hosts in the ecosystem in terms of computation load. In this type of mobility, the destination host's main agent is responsible to confirm the mobility, enabling the consideration of existing constraints, such as communication, or physical resources available. The Virtual Mobility Mechanism, more important in the context of this methodology, enables agents to move to other agent communities (e.g., energy retailers) deployed on the same physical host, in order to take use of their services, interact with other agents, or get access to shared resources at the destination entity (e.g., citizen energy community). Agents that make use of this type of mobility can engage in aggregation entities that are a good fit for their profiles, bringing them closer to realizing the full potential of their energy resources. The primary distinction between virtual and physical mobility is that virtual mobility occurs within the same physical host, eliminating the need for the agent to restart its execution. In the context of this work, an agent can, for instance, enter both RL training ACOMs, and perform a training with few iterations to understand which algorithm best suits its profile, and from there, move to the ACOM that will bring it better results.

### Reinforcement learning training

The reinforcement learning training in the proposed methodology focuses on two main blocks, the environment and the agents. The environment incorporates the P2P model used and provides agents with customized observations for each one. The agents receive the observations from the environment, compute the action to take, determined by the policy or exploration mechanism, and then execute the after market phase. The architecture of the methodology is shown in Fig. 3. As can be seen, although there are several agents, with actions decided by themselves, these are centralized when entering the environment and the P2P market, in order to guarantee the integrity of the environment. After training, only the policies developed by each agent are returned to A4SG agents.

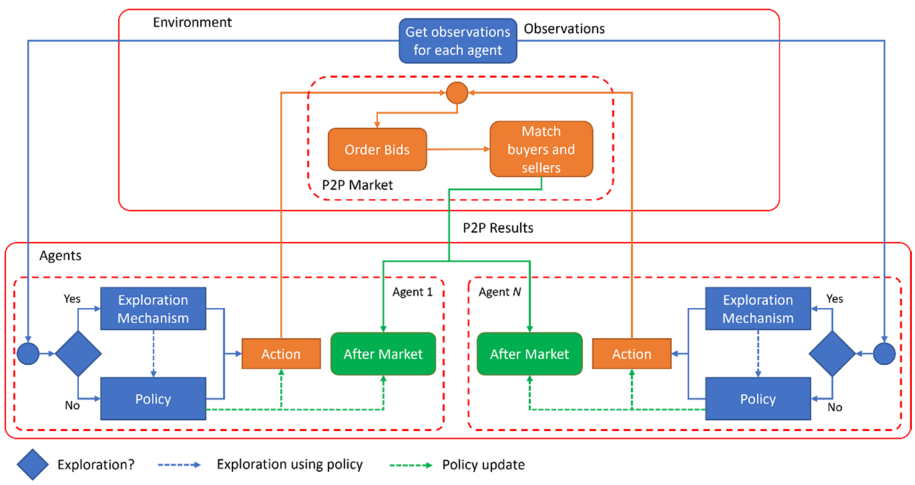


**Fig. 3** RL environment and agents interaction in training

Regarding RL, both algorithms, i.e., TD3 and DDPG, will be used under the same conditions, that is, with the same types of observations, actions and rewards calculated in the same way. Regarding the observation of the state of an agent, this includes several important factors for the decisions of the players when participating in the market. The observation for player $p$ in period $t$ is given by:

$$o_t^p = (Forecast_t^p, Flexibility_t^p, Transactions_{t-1}^p, PeriodTime_t, ToU_t, FiT_t) \tag{2}$$

where $Forecast_t^p$ is the demand forecast of player $p$ for period $t$, in kWh, $Flexibility_t^p$ is the forecasted flexibility of player $p$ for period $t$, also in kWh, $Transactions_{t-1}^p$ is the list of transaction made by player $p$ in period $t-1$ in the P2P market, including information about the price and quantities of energy transacted, and $PeriodTime_t$ provides information about the period of the day represented by period $t$.

The agents' actions are related to the strategy that each one of them develops to participate in the P2P market. Thus, each agent generates two different actions, on regarding the price, and the other regarding the amount of energy to transact, both in the interval [0, 1], representing a percentage value. Thus, the actions of each agent are given by:

$$a_t^p = (aPrice_t^p, aQuantity_t^p) \tag{3}$$

where $aPrice_t^p$ represent the action relative to the price to pay for energy, $aQuantity_t^p$ is the action that indicates the amount of energy to trade in the P2P market, both represented in percentual points, regarding period $t$ and player $p$.

Regarding the proposed exploration mechanism, two types of exploration are used, in order to create a greater range of actions considered. The exploration mechanism is activated from a completely random value, generated in the interval [0,1]. If the value is lower than 0.8, then the actions that were chosen according to the policy are applied without any change. If the value is equal or higher than 0.8 and lower 0.9, then exploration with gaussian noise is activated. And finally, if the value is equal or higher than 0.9 then completely random values are used for all actions. Noise exploration explores actions values relatively close to ideal according to the policy, and random explores any value within the considered ranges.

The truth is that the actions that the policy or the mechanism of exploration of agents generate, for that reason alone, do not have much meaning. In this way, agents must frame these actions in their context, to generate the offers and the strategy to participate in the market. Regarding the price, so that both buyers and sellers feel motivated to participate, the prices offered/asked are limited to the purchase and sale price of energy on the grid. In what regards the energy amount to transact the agents consider the forecast error, and as that, the first step is to determine the potential error in a period. This error is computed using the evaluation metrics of the algorithm for forecasting at the time of testing. If it is the Mean Absolute Percentage Error (MAPE), it must be multiplied by the forecasted value for the period in question; if it is the Mean Absolute Error (MAE), it is used as the error's direct value. After calculating the error, the value of $aQuantity_t^p$ is applied within the forecast's possible range. As such, the price to pay and the amount of energy to be transacted are given by the following equations:

$$BidPrice_t^p = aPrice_t^p * (ToU_t - FiT_t) + FiT_t \tag{4}$$

$$Error_t^p = \begin{cases} MAPE * Forecast_t^p, ifMetric = MAPE \\ MAE, ifMetric = MAE \end{cases} \tag{5}$$

$$\begin{aligned} BidQuantity_t^p = & aQuantity_t^p * ((Forecast_t^p + Error_t^p) \\ & - (Forecast_t^p - Error_t^p)) + (Forecast_t^p - Error_t^p) \end{aligned} \tag{6}$$

where $BidPrice_t^p$ represents the price to pay in the P2P market, in EUR/kWh, $Error_t^p$ is the mean error of the forecast of the player, in kWh, and $BidQuantity_t^p$ is the amount of energy to transact in the P2P market, in kWh, all regarding player $p$ in period $t$.

Bearing in mind that in this methodology the hour-ahead market is used, there is a need of energy forecasts models to carry out the market, and not real values. Therefore, the true impact can only be measured in the period after the transactions are carried out, when the real values of consumption and generation are known. Thus, as shown in Fig. 4, in period $t$ the data related to the transactions carried out are stored, while in period $t + 1$, interactions with the grid to buy or sell energy are carried out, and the reward for period $t$ is calculated.

Regarding the calculation of the reward, it is directly linked to the savings made by the player with the participation in the P2P market. The first step is to calculate the cost or profit of buying or selling the energy to the grid (i.e., $CostGrid_t^p$), where the actual demand of the player is multiplied by the corresponding market price, represented in Eq. (7). Then, the next step is to calculate the money transacted in the P2P market (i.e., $CostMarket_t^p$), that is given by the summatory of price multiplied by energy transacted in each deal of the market, as represented in Eq. (8).

$$CostGrid_t^p = Demand_t^p * \begin{cases} Price_t^{Buy}, if\ Role_t^p = Buyer \\ Price_t^{Sell}, if\ Role_t^p = Seller \end{cases} \tag{7}$$

$$CostMarket_t^p = \sum_{i=0}^{N} (TransactedEnergy_i * Price_i) \tag{8}$$

Even with market transactions, the interaction with the grid to buy/sell energy from/to the grid is almost inevitable. This is because, using the forecast as a basis for the amount of

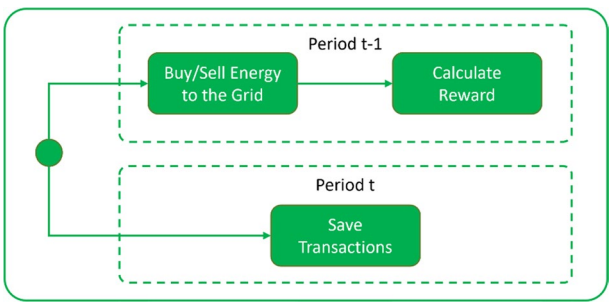


**Fig. 4** RL after market phase

energy to be transacted, there will always be errors, even if small, that make this interaction mandatory. The amount of energy to buy/sell from/to the grid (i.e., $EnExtra_t^p$) is given by the Eq. (9) and is the difference between the real demand and the amount of energy traded in the market. In order to try to reduce the cost of interacting with the grid, when it is necessary to buy, that is, when a buyer does not transact enough energy in the market, or when a seller transacts more energy, flexibility is used to reduce costs. Equations (10) and (11) describe the process of calculating how much flexibility is needed, and the cost associated with this interaction in the after market phase. In Eq. (10) the amount of flexibility is given by the minimum between $Flexibility_t^p$ and $EnExtra_t^p$, both regarding player $p$ in period $t$, in kWh. On the other hand, the result of Eq. (9) and (10) are used to calculate the cost of buy/sell energy to grid in the after market phase. The flexibility is used only in the periods that demands the buying of more energy from the grid.

$$EnExtra_t^p = \sum_{i=0}^{N} (TransactedEnergy_i) - Demand_t^p \tag{9}$$

$$UsedF_t^p = \min(EnExtra_t^p, Flexibility_t^p) \tag{10}$$

$$CostExtra_t^p = \begin{cases} EnExtra_t^p * FiT_t, if\, Role_t^p = BuyerANDEnExtra_t^p \geq 0 \\ EnExtra_t^p * FiT_t, if\, Role_t^p = SellerANDEnExtra_t^p < 0 \\ (EnExtra_t^p - UsedF_t^p) * ToU_t, if\, Role_t^p = SellerANDEnExtra_t^p \geq 0 \\ (EnExtra_t^p - UsedF_t^p) * ToU_t, if\, Role_t^p = BuyerANDEnExtra_t^p < 0 \end{cases} \tag{11}$$

Finally, the reward is calculated by measuring the impact of participation in P2P in reducing costs or increasing profits, so there is a differentiation in the formula for sellers and buyers. The equation that gives the reward is then given by:

$$r_t^p = \begin{cases} CostGrid_t^p - CostMarket_t^p + CostExtra_t^p, if\, Role_t^p = Buyer \\ CostMarket_t^p - CostGrid_t^p - CostExtra_t^p, if\, Role_t^p = Seller \end{cases} \tag{12}$$

In order to facilitate the development and integration of the methodology in the A4SG ecosystem, the OpenAI Gym toolkit (Brockman et al. 2016) and the Ray RLib library (Liang et al. 2017) were used. The OpenAI Gym toolkit enables research, development, and application of RL. It integrates a large number of well-known tasks that expose a common interface that allows direct comparison of the performance results of various RL algorithms. In addition, the environments that follow the OpenAI Gym settings and requirements are often efficient in training processes that involve a high number of iterations. The Ray RLlib library provides the implementation of several RL algorithms, such as the one used in the proposed methodology in this paper, i.e., DDGP and TD3. If the environments where the algorithms are applied are OpenAI Gym-compliant, then the integration between the two libraries is quite straightforward since the agents that this library provides already allow and aim to make this connection.

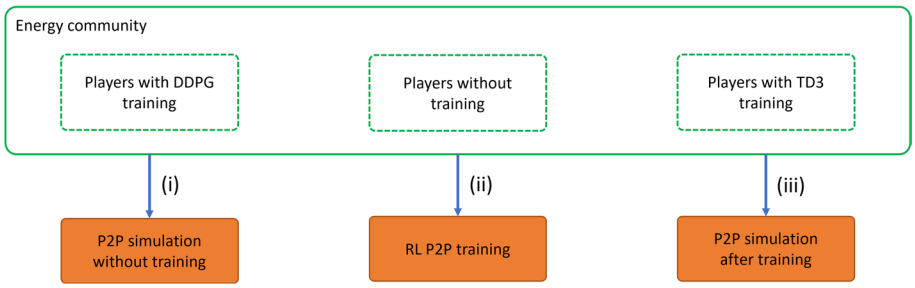


**Fig. 5** Case study steps

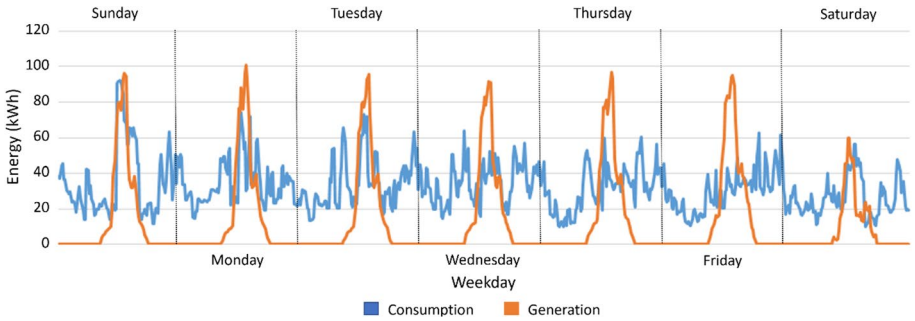


**Fig. 6** Community general demand

## Case study

In order to test the proposed methodology, a case study was developed using an energy community of 50 players (Goncalves et al. 2022). The objective was to validate whether the two reinforcement learning algorithms used allow A4SG ecosystem's agents to improve their participation in the P2P energy transactions market. From the 50 players of the energy community, 20 will be sent to training with TD3, 20 to training with DDPG, and 10 will remain without training, in order to have a strong component of comparison between these three aspects. The distribution of the players is completely random. The flow and steps of the case study are shown in Fig. 5. The objective is to simulate a P2P week without training (step i) train the agents with the proposed RL models (step ii), and then compare the simulated market week with the agents' participation in the market with the trained policies (step iii). From the point of view of the A4SG ecosystem, the agents will be distributed among the different training ACOMs, despite meeting together in the ACOM of the energy community.

In order to test the application of the methodology in a real context, the energy community used was created from real player profiles. These players are all residential, thus having relatively similar capacities in terms of generation. In Fig. 6 can be seen the consumption and general generation of the community throughout the studied week.

Although the TD3 algorithm is quite robust as far as hyperparameters are concerned, and as such, no tuning is necessary, DDPG is subject to this issue. However, both algorithms were used with the same hyperparameters, which are shown in Table 1, in order to be able to make a direct comparison between the two.

**Table 1** TD3 and DDPG training hyperparameters

| Hyperparameter | Value |
|---|---|
| Train batch size | 100 |
| Tau (τ) | 0.005 |
| Gamma (γ) | 0.99 |
| Critic learning rate | 0.001 |
| Actor learning rate | 0.001 |
| Policy delay | 2 |
| Policy noise | 0.2 |
| Noise clip | 0.5 |

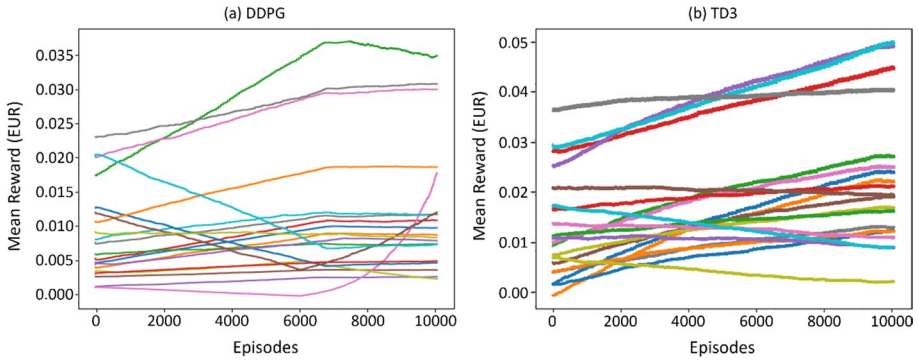


**Fig. 7** Mean reward for each agent per training episode: (**a**) DDPG, and (**b**) TD3

In order to compare the performance of P2P agents with and without training, three negotiation profiles were used prior to RL training to generate offers:

- Greedy profile: agents adhering to this profile will attempt to trade between 80 and 100% of the forecasted demand at a price between 20 and 80% within market limits;
- Safety profile: where agents attempt to negotiate between 90 and 110% of the forecast, while in terms of price, they use the limits between 60 and 100%;
- Cheaper profile: where agents try to seek deals at a lower price. These contracts aim to trade between 80 and 100% of the forecast but utilize only between 0 and 50% of the price limits.

## Results

The most effective method for assessing the performance of reinforcement learning algorithms is to examine agent rewards across training iterations. An increase in the value of rewards indicates that agents are performing better at the task for which they are undergoing training. However, in competitive contexts such as the P2P market, it is common for some agents to achieve better results than others, as strategies are not shared and each agent seeks to achieve the best results for himself, which in most cases results in poorer outcomes for the others. Figure 7 depicts the average reward received

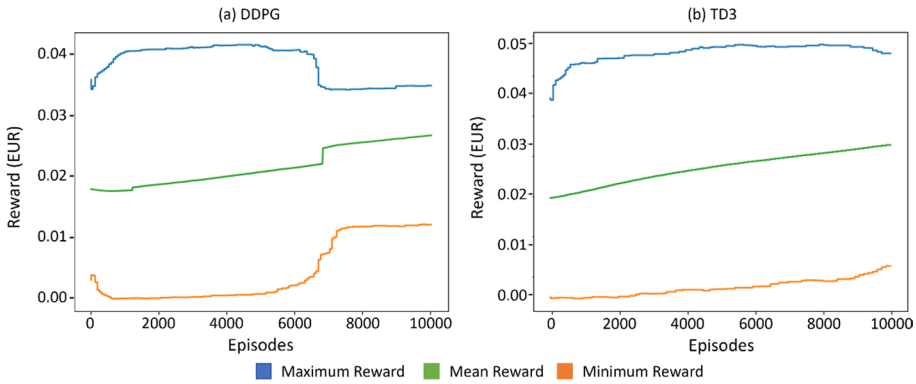


**Fig. 8** Maximum, mean, and minimum reward per training episode: (**a**) DDPG, and (**b**) TD3

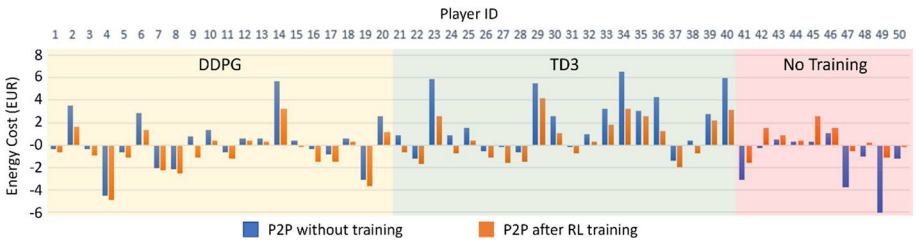


**Fig. 9** Comparison of the energy costs during a week before and after the RL training, for both RL algorithms and the agents without training

by each agent for each algorithm throughout the training episodes. In general, the TD3 algorithm enabled a better learning for most agents, whereas roughly half of the agents in the DDPG were very close to their initial values. In addition, agents with superior learning achieved better results in the TD3, saving an average of 0.05 EUR per period of market share, whereas in the DDPG, this value was close to 0.03 EUR.

From the perspective of the general community, there are greater differences in terms of rewards between the two algorithms. The minimum, mean, and maximum are depicted in Fig. 8 for the agents that participated in the training of each algorithm. On the one hand, the TD3 has a much higher maximum reward value than the DDPG, with a value of 0.47 compared to 0.34 for the DDPG. Regarding the average value, the behavior of both was quite similar, as the difference was only 0.02; however, the TD3 again held the advantage. On the other hand, as far as the minimum value is concerned, the DDPG has the advantage, and based on the analysis of the graph, there appears to be a greater balance between all market-participating agents. This indicates that, with the use of DDPG and a competitive strategy, a balance has been reached between the players' profits.

Considering the overall savings of the community and the fact that the rewards are positive for all agents who participated in the training, it is possible to conclude that costs were reduced, i.e., the players saved money through the training that enabled them to develop P2P participation strategies. Figure 9 depicts the results of participating in a P2P week before and after training, broken down by agents who trained with DDPG, agents who trained with TD3, and agents who received no training. As can be seen, the most significant difference is between agents with and without

training, with agents with trained policies having a significant advantage in the week following training. From a total profit of 13.29 EUR to a loss of 3.78 EUR, untrained agents incurred a loss. Comparing trained agents, those with DDPG saved 15.99 EUR (i.e., from a cost of 7.38 EUR to a profit of 8.61 EUR, representing savings of 217%), while those with TD3 saved 28.66 EUR (i.e., from a cost of 36.64 EUR to a cost of 7.97 EUR, representing savings of 78%). These results demonstrate that agents trained with the TD3 algorithm realized greater cost savings. Analyzing the results, in what regards the percentage, the DDPG had better results, but this is due to the low cost previously associated with the players who trained with this algorithm, going from having costs to profit.

The actions of agents are an additional topic worthy of investigation. As depicted in Fig. 10a, substantial changes in the mean of the agents' actions were not required for the agents to learn and increase their rewards. The truth, however, is that around episode 7500, both buyers and sellers began to acquire lower-priced stocks. This is due to different approaches by agents, including: (i) in general, agents found a better way to deal with forecast error when selecting the amount of energy to trade, and (ii) sellers began to realize that lowering the price of energy sales could result in a greater profit, as it would result in the sale of more energy. Figure 10b depicts the average actions of the agents during the P2P week following RL training. The first conclusion that can be drawn is that when there is less generation, that is, the first and last periods where there is a transaction (usually between 8:00 am/9:00 am and 5:00 pm/6:00 pm), sellers reduce their asking price and increase the quantity of energy to be traded. This is because, as energy quantities decrease, sellers attempt to sell as much as possible at a low price in order to attract buyers. As far as buyers are concerned, their behavior is more consistent, as they attempt to find a strategy in which they purchase only the amount of energy they require at a compensating price.

The final metric to examine is the amount of energy traded on the P2P market. As a matter of fact, it can also be used as an evaluation metric for the participation of agents in the market as a community, since the goal is to maximize the energy transacted in order to create greater sustainability within the energy community while interacting with the utility grid as little as possible. Figure 11 shows the energy

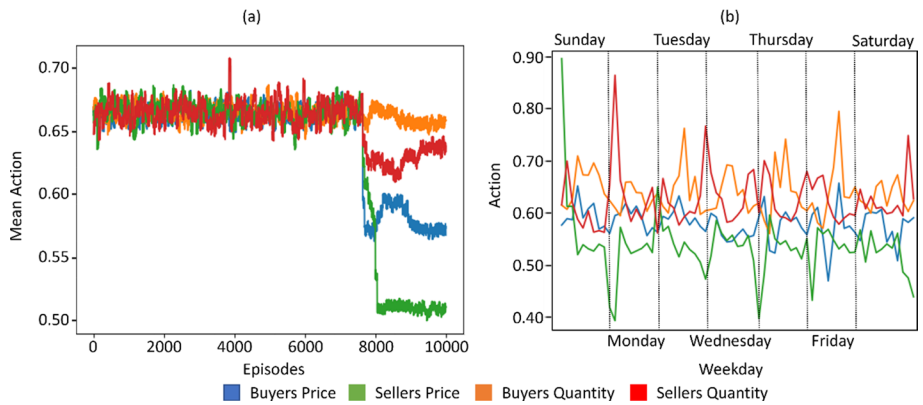


**Fig. 10** Actions chosen by agents: (**a**) Average of actions throughout the training (**b**) Average of actions chosen in the post-training week

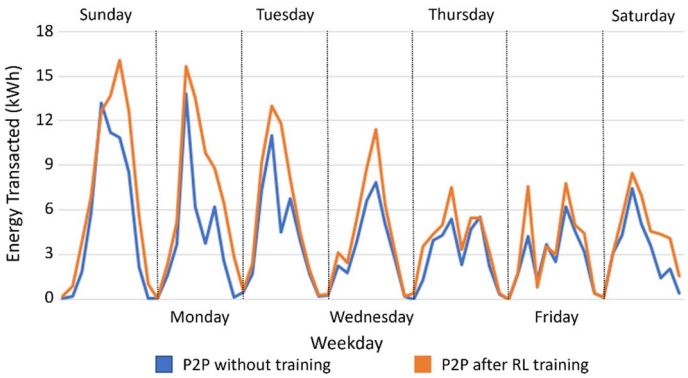


**Fig. 11** Energy transacted in the P2P market before and after the RL training

exchanged per period in the week preceding and following RL training. As can be seen, in the vast majority of time periods following training, the amount of energy transferred increased by 40%, from 245.11 to 342.84 kWh, getting much closer to the maximum value throughout the week (i.e., 387.39 kWh).

The results obtained are highly positive, particularly when comparing to the performance of trained and untrained players. It is apparent that a player trained with one of the RL models can make use of the intelligence it provides to employ the best strategies in each situation to minimize their energy costs. In addition, since training is conducted in a competitive manner with other players, it is possible to perceive multiple strategies and prepare the player for various market circumstances. The proposed methodology considers the forecast error to deal with uncertainty, being an improvement when compared, for instance, to the model proposed in Qiu et al. (2021a), where forecast errors are not considered. Also, the case study of this paper used a realistic dataset, with real measurements, for a community of 50 agents, contributing to the testing of the scalability of the used algorithms. The dataset used is a public dataset that can be used by other authors to compare results (Goncalves et al. 2022).

## Conclusions

The main objective of the proposed methodology is to improve the participation of energy players in P2P markets and local energy markets. In addition, the methodology was integrated into an agent-based ecosystem, in order to facilitate its use, and to make a direct connection with existing multi-agent systems. The methodology showed positive results in terms of reducing the costs of players who train with reinforcement learning, especially when compared to players without training. Each algorithm trained 20 agents, where the deep deterministic policy gradient (DDPG) reduced costs by 15.99 EUR (representing savings of 217%), i.e., on average 0.80 EUR per agent in one week, and the twin delayed DDPG (TD3) reduced costs by 28.07 EUR (representing savings of 78%), i.e., on average 1.43 EUR per agent in a week.

In addition to having good results from an individual point of view, this methodology can also benefit aggregation entities, such as energy communities. This is because, when compared to a week of peer-to-peer without training, after using the methodology, it allowed to increase the energy transacted by 40%, and significantly approaching the maximum possible according to the consumption and generation profiles of the considered players.

The integration of this type of RL methodologies in agent-based systems, which have a greater proximity to real contexts, enables RL models to be applied to real data more easily and to test their application in different types of players who may come to use these models to improve their active participation in the smart grid. The work proposed in this paper is susceptible to continual development through the inclusion of new variables that enhance the agent's environment perception and behavior in the market. For instance, the use of flexibility is already addressed in this study, but only in the after-market phase, whereas its use may be explored to determine the quantity of energy to be transacted, so that a portion of the agents can participate in the market with greater flexibility.

**Abbreviations**

| | |
|---|---|
| A4SG | Agent-based ecosystem for smart grid modelling |
| ACOM | Agent Community |
| CECOS | Citizen Energy Communities Operator System |
| DDPG | Deep Deterministic Policy Gradient |
| DA | Double-auction |
| DER | Distributed energy resources |
| DRL | Deep reinforcement learning |
| FiT | Feed-in-tariff |
| LEM | Local energy market |
| MAE | Mean absolute error |
| MAPE | Mean absolute percentage error |
| MARL | Multi-agent reinforcement learning |
| MAS | Multi-agent system |
| P2P | Peer-to-peer |
| PEAK | Python-based agent communities ecosystem |
| RES | Renewable energy sources |
| RL | Reinforcement learning |
| SPADE | Smart Python Agent Development Environment |
| TD3 | Twin delayed deep deterministic policy gradient |
| TE | Transactive energy |
| ToU | Time-of-use |

**Acknowledgements**
The authors acknowledge the work facilities and equipment provided by GECAD research center (UIDB/00760/2020) to the project team.

**About this supplement:** This article has been published as part of *Energy Informatics Volume 5 Supplement 4, 2022: Proceedings of the Energy Informatics.Academy Conference 2022 (EI.A 2022)*. The full contents of the supplement are available online at https://energyinformatics.springeropen.com/articles/supplements/volume-5-supplement-4.

**Author contributions**
All the authors made contributions to the conception of the proposed solution. The architectural design and software developments were mainly done by HP, and LG. Data acquisitions and analysis were done by HP. The interpretation of the data was done by all the authors. The first draft was written by HP, and LG, while all other authors contributed to the final version of the paper. All authors read and approved the final manuscript.

**Funding**
This work has received funding from FEDER Funds through COMPETE program and from National Funds through (FCT) under the project PRECISE (PTDC/EEI-EEE/6277/2020).

**Availability of data and materials**
The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

## Declarations

**Ethics approval and consent to participate**
Not applicable.

**Consent for publication**
Not applicable.

**Competing interests**
The authors declare that they have no competing interests.

Accepted: 11 October 2022
Published: 21 December 2022

**References**

Abrishambaf O, Lezama F, Faria P, Vale Z (2019) Towards transactive energy systems: an analysis on current trends. Energy Strateg Rev 26:100418
Arulkumaran K, Deisenroth MP, Brundage M, Bharath AA (2017) Deep reinforcement learning: a brief survey. IEEE Signal Process Mag 34:26–38
Botvinick M, Ritter S, Wang JX, Kurth-Nelson Z, Blundell C, Hassabis D (2019) Reinforcement learning, fast and slow. Trends Cogn Sci 23:408–422
Brockman G, Cheung V, Pettersson L, Schneider J, Schulman J, Tang J, Zaremba W (2016) OpenAI Gym.
Chen T, Bu S (2019) Realistic peer-to-peer energy trading model for microgrids using deep reinforcement learning. Proc 2019 IEEE PES Innov Smart Grid Technol Eur ISGT-Europe 2019. https://doi.org/10.1109/ISGTEUROPE.2019.8905731
Chen YC, Liu HM (2021) Evaluation of greenhouse gas emissions and the feed-in tariff system of waste-to-energy facilities using a system dynamics model. Sci Total Environ 792:148445
Chen T, Bu S, Liu X, Kang J, Yu FR, Han Z (2022) Peer-to-peer energy trading and energy conversion in interconnected multi-energy microgrids using multi-agent deep reinforcement learning. IEEE Trans Smart Grid 13:715–727
Chicco G, Somma M Di, Graditi G (2021) Overview of distributed energy resources in the context of local integrated energy systems. Distrib Energy Resour Local Integr Energy Syst Optim Oper Plan 1–29
Chiu WY, Hu CW, Chiu KY (2022) Renewable energy bidding strategies using multiagent Q-learning in double-sided auctions. IEEE Syst J 16:985–996
de São JD, Faria P, Vale Z (2021) Smart energy community: a systematic review with metanalysis. Energy Strateg Rev 36:100678
Dudjak V, Neves D, Alskaif T et al (2021) Impact of local energy markets integration in power systems layer: a comprehensive review. Appl Energy 301:117434
Friedman D (2018) The double auction market: institutions, theories, and evidence. Routledge
Fujimoto S, Van Hoof H, Meger D (2018) Addressing function approximation error in actor-critic methods. 35th Int Conf Mach Learn ICML 2018 4:2587–2601
Gomes L, Morais H, Gonçalves C, Gomes E, Pereira L, Vale Z (2022) Impact of forecasting models errors in a peer-to-peer energy sharing market. Energies 15:3543
Goncalves C, Barreto R, Faria P, Gomes L, Vale Z (2022) Dataset of an energy community's consumption and generation with appliance allocation for one year. https://doi.org/10.5281/ZENODO.6778401
Gronauer S, Diepold K (2021) Multi-agent deep reinforcement learning: a survey. Artif Intell Rev 55:895–943
Gržanić M, Capuder T, Zhang N, Huang W (2022) Prosumers as active market participants: a systematic review of evolution of opportunities, models and challenges. Renew Sustain Energy Rev 154:111859
Liang E, Liaw R, Moritz P, Nishihara R, Fox R, Goldberg K, Gonzalez JE, Jordan MI, Stoica I (2017) RLlib: abstractions for distributed reinforcement learning. 35th Int Conf Mach Learn ICML 2018 7:4768–4780
Mota B, Albergaria M, Pereira H, Silva J, Gomes L, Vale Z, Ramos C (2021) Climatization and luminosity optimization of buildings using genetic algorithm, random forest, and regression models. Energy Inform. https://doi.org/10.1186/s42162-021-00151-x
Padakandla S, Bhatnagar KJP (2020) Reinforcement learning algorithm for non-stationary environments. Appl Intell 50:3590–3606
Palanca J, Terrasa A, Julian V, Carrascosa C (2020) Spade 3: supporting the new generation of multi-agent systems. IEEE Access 8:182537–182549
Pereira H, Gomes L, Faria P, Vale Z, Coelho C (2021) Web-based platform for the management of citizen energy communities and their members. Energy Inform. https://doi.org/10.1186/s42162-021-00155-7
Qiu D, Wang J, Wang J, Strbac G (2021a) Multi-agent reinforcement learning for automated peer-to-peer energy trading in double-side auction market. IJCAI Int Jt Conf Artif Intell 3:2913–2920
Qiu D, Ye Y, Papadaskalopoulos D, Strbac G (2021b) Scalable coordinated management of peer-to-peer energy trading: a multi-cluster deep reinforcement learning approach. Appl Energy 292:116940
Recht B (2019) A tour of reinforcement learning: the view from continuous control. Annu Rev Control Robot Auton Syst 2:253–279
Reis FGI, Gonçalves I, Lopes ARM, Henggeler Antunes C (2021) Business models for energy communities: a review of key issues and trends. Renew Sustain Energy Rev 144:111013
Samende C, Cao J, Fan Z (2022) Multi-agent deep deterministic policy gradient algorithm for peer-to-peer energy trading considering distribution network constraints. Appl Energy 317:119123
Venizelou V, Philippou N, Hadjipanayi M, Makrides G, Efthymiou V, Georghiou GE (2018) Development of a novel time-of-use tariff algorithm for residential prosumer price-based demand side management. Energy 142:633–646
Wu Y, Wu Y, Guerrero JM, Vasquez JC (2021) Digitalization and decentralization driving transactive energy Internet: Key technologies and infrastructures. Int J Electr Power Energy Syst 126:106593

## Publisher's Note