Received: 9 October 2025 | Revised: 14 January 2026 | Accepted: 25 March 2026 | Published online: 11 May 2026

RESEARCH ARTICLE

Journal of Computational and Cognitive Engineering
2026, Vol. 5(2) 187–201
DOI: 10.47852/bonviewJCCE62027858

BON VIEW PUBLISHING

Multi-agent Reinforcement Learning with Clustering and Forecasting for Optimized Energy Sharing in Microgrids

Daswin De Silva[1,*], Thimal Kempitiya[1], Nuwan Madhusanka[1], Prabod Rathnayaka[1], Nishan Mills[1], Andrew Jennings[1] and Milos Manic[2]

[1] *La Trobe Artificial Intelligence Institute, La Trobe University, Australia*

[2] *Department of Computer Science, Virginia Commonwealth University, USA*

Abstract: The increasing prevalence of renewable energy and the evolving nature of energy consumption have motivated the need for more complex and dynamic microgrid energy management systems. Recent advances in artificial intelligence (AI) address these challenges by learning, predicting, and optimizing based on the large volumes of data generated by microgrid systems and related operations. Drawing on this context, the article proposes a novel framework for multi-agent reinforcement learning (MARL) with clustering and forecasting for optimized energy sharing in a microgrid environment with renewables and battery storage integration. The framework consists of three components: first, a structure-adapting unsupervised learning approach for creating clusters of prosumer energy consumption and generation patterns; second, a time-series forecasting ensemble for predicting future behaviors of the prosumers; and third, a continuous internal auction with a MARL for optimized energy sharing within the microgrid that collectively leads to reduced dependence on external energy sources. The proposed framework is empirically evaluated in the microgrid setting of a large multi-campus tertiary education institution. The results of this evaluation include stabilization of mean reward gain between independent agent and multi-agent models, impact of forecasting on MARL across seasonal variation, performance gains of 10–15% of MARL against heuristics, and scalability of the framework against cost, stability, reward, and convergence metrics. These results confirm the effectiveness of this AI framework for optimized prosumer energy sharing in microgrids with renewables and battery storage integration.

Keywords: multi-agent reinforcement learning, deep learning, demand forecasting, microgrid auction, artificial intelligence

1. Introduction

Microgrids are an innovative "system approach" that constitutes energy consumption loads, renewables generation and storage options as a subsystem of the overall energy grid [1, 2]. Microgrid energy management systems (MEMS), distributed energy resources, prosumer behaviors, demand response, electric vehicles, and local controllers are recent developments that have further expanded microgrid operations toward achieving financial, environmental, and organizational objectives [3, 4]. However, this also means that microgrid operations are increasingly more complex and challenging. The primary challenge in current microgrids composed of prosumers is the bidirectional energy flow, which adds complexity to integration with conventional systems. This increases the complexity of maintaining the microgrid's topology and maintaining power balance and voltage fluctuations. The second challenge arises from the variability and uncertainty in the microgrid, which stems from dynamic load profiles and renewable energy sources. This could lead to increased complexity in energy storage management and dynamic load balancing. This complexity has led to the adoption of artificial intelligence (AI) models for the optimized operation of microgrids [5]. Despite recent work in supervised learning for forecasting generation and consumption and in reinforcement learning for optimizing renewable generation, a unified approach for optimized energy sharing and management of distributed energy resources in a microgrid setting is lacking. In this article, we address this gap by proposing a novel AI framework based on the following three research contributions, (1) structure-adapting unsupervised learning approach for generating clusters of past consumption and generation behaviors of prosumers, (2) time-series forecasting ensemble for predicting future behaviors of the prosumers, and (3) continuous internal auction with multi-agent reinforcement learning (MARL) for optimized energy sharing. The structure-adapting unsupervised learning approach generates clusters of prosumer consumption and generation behaviors that are utilized to identify the optimum battery storage distribution, predict behaviors of each prosumer cluster, and optimize energy sharing. We evaluate the learning capabilities of this AI framework in the real-world microgrid setting of a large multi-campus tertiary education institution. The rest of the paper is organized as

***Corresponding author:** Daswin De Silva, La Trobe Artificial Intelligence Institute, La Trobe University, Australia. Email: d.desilva@latrobe.edu.au

© The Author(s) 2026. Published by BON VIEW PUBLISHING PTE. LTD. This is an open access article under the CC BY License (https://creativecommons.org/licenses/by/4.0/).

Table 1
Terms and abbreviations used in this article

Symbol	Type	Description
A_{id}	Int	Id of an agent
$E_{id,t}$	kWh	Amount of energy sold/bought by agent A_{id} on hour t. Negative values mean buy, and positive values mean sell
$P_{id,t}$	Currency/kWh	Price at which agent A_{id} bids on hour t
$E_{buy,t}$	kWh	The sum of the amount of energy from buying bids on hour t
$E_{sell,t}$	kWh	The sum of the amount of energy from accepted selling bids on hour t. This value starts at 0 and is updated during the auction
Bid_t	$[E_{id}, P_{id}]$	Bid of agent A_{id} at hour t
P_t	Currency/kWh	Price of electricity from the grid at hour t
$Sell\_bids_t$	Array of type	Bid_t Sell bids of all of the agents at hour t, sorted in ascending order of price
Buy_bids_t	Array of type Bid_t	Buy bids of all of the agents at hour t, sorted in ascending order of price
$Sell\_orders_t$	Array of type Bid_t	Accepted sell bids of all of the agents at hour t, sorted in ascending order of price
Buy_orders_t	Array of type Bid_t	Accepted buy bids of all of the agents at hour t, sorted in ascending order of price

follows. Section 2 articulates the proposed framework for MARL with clustering and forecasting for optimized prosumer energy sharing, followed by Section 3, which presents the experiments and results, and Section 4, which concludes the paper. Table 1 presents terms and abbreviations used in this article.

2. Related Work

MEMS are responsible for the effective operation of microgrids. Starting with classical systems, meta-heuristic methods, stochastic programming, and model predictive control (MPC) methods, MEMS have gradually evolved into the robust application of AI models and approaches to address the increasing complexity, variability, and uncertainty of microgrids. This section will focus on related work in unsupervised, supervised, and reinforcement learning that is relevant to the proposed framework.

Unsupervised learning algorithms learn a structure or representation from unlabeled datasets. These structures are frequently referred to as clusters, profiles, or segments. The energy domain generates many high-velocity and high-frequency unlabeled datasets from smart meters, control systems, and grid management systems. Using smart meters attached to houses, buildings, machinery, and geographical areas, several unsupervised learning approaches have been proposed to generate clusters of consumption and/or generation profiles. Czétány et al. [6] used a clustering-based method to profile the energy consumption of households, using k-means, fuzzy k-means, and agglomerative hierarchical clustering on daily and annual energy data to generate the consumption profiles. Zhan and Chong [7] used principal component analysis for feature extraction from energy data and to generate clusters. Peter et al. [8] conducted a review of intelligent protection schemes implemented in AC, DC, and AC/DC hybrid microgrids, alongside their limitations, protective features, and performance evaluation. A deep autoencoder approach for energy theft detection in microgrid settings [9] and cyber–physical anomaly detection in inverter-based microgrids [10] have also been proposed.

Supervised learning algorithms learn a function between input attributes and a target attribute/s, using labeled data. Given the temporal nature of energy production, supply, consumption, and management, time-series data streams generated by energy infrastructure and equipment can be efficiently leveraged as labeled data for the prediction and forecasting of future behaviors of consumption and generation. Supervised learning has been effectively applied in energy use cases, such as price, load forecasting, real-time monitoring, and anomaly detection [11–13]. Time-series forecasting has also been used to handle anomalies in data, including missing values, and to develop efficient and scalable load forecasting in distributed and causality processing scenarios [14]. Hybrid approaches of combining unsupervised learning with supervised learning have also been proposed [15].

Several recent studies have applied reinforcement learning to optimize the operation of microgrids with smart loads, generation, and energy storage. These applications need to consider special characteristics of microgrids, such as power flow constraints at the point of common coupling to the utility grid [16, 17], the option for microgrid internal demand response markets [18], existing energy trading possibilities offered by power utilities [19], and the eventual emergence of novel markets designed for a microgrid dominated power system [20]. However, optimization with reinforcement learning utilizes only a subset of relevant AI techniques. In a recent taxonomy of AI applications for distributed energy resources, Sierla et al. [21] investigated the applications of supervised, unsupervised, and reinforcement learning, as well as combinations of these techniques [22].

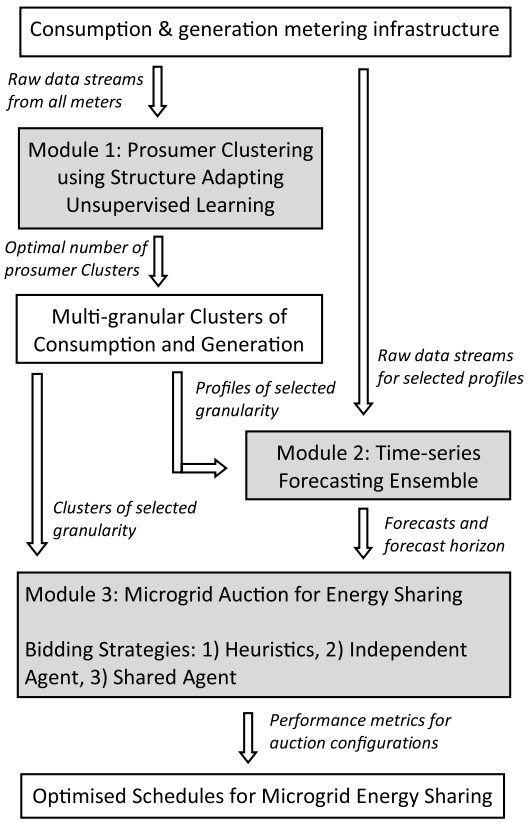
The proposed AI framework aims to address these limitations as follows. As battery storage is an expensive and scarce resource in a microgrid setting [23], the prosumer clusters ensure this limited resource is utilized in the most effective manner [24, 25]. The time-series forecasting ensemble predicts future energy use for both prosumer clusters and individual participants.

It also aims to address the uncertainties that occur in the energy sharing from external factors such as weather and consumer behavior [26, 27]. The continuous internal auction lays the foundation for efficient allocation of surplus in the microgrid [28] and facilitates competitiveness among microgrid nodes for balanced energy usage [29]. MARL [30] enables each prosumer to operate independently in the continuous internal auction and thereby automates the scaling and optimization of energy sharing in the microgrid [31]. This delivers a financial benefit for prosumers considering fairness and autonomy [32], as well as decreased energy usage from the external grid, which then also enables carbon emissions reduction.

3. Methodology

This section delineates the structure of the proposed AI framework and its composition in terms of the following three core modules: (1) prosumer clusters: structure-adapting unsupervised learning approach for generating clusters of prosumer usage and generation patterns from raw data streams; (2) time-series forecasting ensemble: an ensemble of supervised learning algorithms for forecasting consumption and generation; and (3) microgrid auction with multi-agent bidding: a continuous internal auction based on MARL for optimized microgrid energy sharing. Figure 1 illustrates the structure and composition of this AI framework. It begins with the extraction and processing of consumption and generation data from the metering infrastructure, as well as other devices and equipment, at the highest granularity available (typically 5–15 min frequencies). The inherently spatiotemporal granularity of energy data is central to developing AI techniques for energy systems. For instance, high-resolution temporal data combined with spatial granularity enables precise modeling of consumption behaviors in energy [33, 34]. In contrast to healthcare data, where fine-grained, individual-level measurements are available, energy data are more challenging due to the complexity of dynamic consumption, diverse device types, and the influence of external factors. Next, the raw data streams are received by Module 1 for the generation of prosumer clusters. The optimal prosumer clusters and multi-granular profiles of consumption and generation are then input into the other two modules. In Module 2, optimum clusters are used to determine the forecast for consumption and generation. In Module 3, cluster information is used to determine the optimal battery allocation for prosumers.

Figure 1
Modular composition and information flow of the proposed framework



Module 2 receives prosumer clusters as well as the raw data streams for forecasting future consumption and generation of all buildings to determine uncertainties that occur due to external behaviors of the microgrid. Forecasting for the prosumer clusters ensures the forecasts are robust, efficient, and scalable, specifically in a microgrid setting where forecasting is aligned to the prosumer clusters instead of individual prosumers. These forecasts are then passed on to Module 3, which initiates a microgrid auction for optimized prosumer participation and competitive energy sharing. The optimum energy-sharing strategy is composed of three heuristic methods and two MARL methods, independent learning, and a shared agent. The following three subsections provide the details of each module.

3.1. Module 1: prosumer clustering using structure-adapting unsupervised learning

The prosumer clusters are generated using a structure-adapting unsupervised learning approach. These clusters are propagated through to Module 2 for cluster-based scalable forecasting and in Module 3 for an optimal battery allocation strategy. The structure-adapting unsupervised learning approach is implemented using the Hyperseed algorithm [35] and the k-means algorithm. Hyperseed is a few-shot unsupervised learning algorithm with a learning rule defined by a single vector operation. It has been shown to learn a complete feature space from a limited number of input vectors within a few training iterations. This few-shot learning capability makes Hyperseed well-suited for low-compute energy sharing in microgrid environments.

Using Hyperseed, we can represent different groupings of prosumers based on past data of consumption and generation behaviors. We represent the diverse prosumer consumption patterns using two sets of variables in terms of time bands and date ranges, leading to a total of 52 statistical features. These features are described in Table 2. We define prosumers and their statistical representation as follows: $C_i = \{s_1, s_2, ..., s_j\}$, where C_i is the ith prosumer and s_j is the jth statistical feature of C_i. Learned prosumer behaviors are further grouped into clusters using the k-means algorithm. The k-means algorithm generates a set of distinct, non-overlapping groups. It operates by iteratively assigning each data point to the nearest cluster center and then updating the cluster centers as the mean of the assigned points. The optimal number of clusters for k-means is determined using the elbow method by analyzing the inflection point of the within-cluster sum of squares. Using these clusters, we define different energy consumption profiles. Hyperseed learns prosumer behaviors, which are then grouped into distinct, non-overlapping groups using k-means.

Table 2
Description of 52 features used in Module 1 prosumer clustering

Feature label	Description	Feature count
Mean energy consumption	Mean hourly energy usage of a given date range	1
Mean energy consumption by month	Mean hourly energy usage of a given date range calculated monthly	12
Mean peak energy consumption	Mean peak energy usage of a given date range	1
Mean peak energy consumption by month	Mean peak energy usage of a given date range calculated monthly	12
Mean working hour energy consumption	Mean hourly energy usage of a given date range calculated separately for working hours and non-working hours	2
Mean working hour energy consumption by month	Mean hourly energy usage of a given date range calculated separately for working hours/non-working hours and each month	24

Once the energy consumption profiles are identified, we use the profiles to determine the allocation of available batteries. If we have n batteries available, with each profile's median consumption being $mC_1, mC_2, ..., mC_j$ for j number of profiles, we calculate the total consumption of all profiles MC. Therefore, allocation for each profile can be presented as Equation (1), with remaining batteries allocated to the highest consumption profile. Equation (1) provides a coarse baseline allocation and does not explicitly model temporal variability or generation intermittency. Its use is restricted to scenarios with relatively stable prosumer profiles.

$$\text{Allocated Batteries for } C_j = \left\lfloor \frac{n \cdot m \cdot C_j}{MC} \right\rfloor \tag{1}$$

3.2. Module 2: time-series forecasting ensemble

The time-series forecasting ensemble generates short-term load and generation forecasts for each individual prosumer and each prosumer cluster. Load and generation forecasts help to reduce the uncertainties arising from external factors, such as seasons, weather events, people's behaviors, and other planned activities. The major challenges of forecasting for prosumers in a microgrid setting are forecast accuracy, missing values and anomalies in individual prosumer data [36], and scalability [37]. This module utilizes an ensemble of supervised learning algorithms to address the challenge of forecast accuracy. The ensemble consists of nine algorithms: random forest; k-Nearest Neighbors; support vector machine; multilayer perceptron with two variations (multi-step dense and single-step dense); XGBoost; convolutional neural network (CNN) with 1-D CNN with 32 filters followed by two fully connected layers with hidden unit size of 32 and 1, respectively; Temporal Convolution Network; and Long Short-Term Memory [38]. This ensemble follows a dynamic leader selection mechanism. At each forecasting step, the model with the lowest recent prediction error over a rolling evaluation window is selected as the leader, and its output is used as the final forecast. This adaptive selection strategy enables the ensemble to respond effectively to nonstationary consumption and generation patterns in an energy-sharing setting by identifying the most reliable predictor under current conditions (Equation (2)). Here, y_t denotes the observed value of the target variable at time step t, and $\hat{y}_t$ represents its forecast.

$$\hat{y}(t) = y^{\left(\widehat{\arg\min_i Err_i(t-1)}\right)}(t) \tag{2}$$

Model performance can be evaluated using a combination of absolute, relative, and task-specific metrics. Absolute measures such as total operational cost and root mean square error (RMSE) quantify raw performance and directly reflect system objectives. To enable scale-invariant comparison across models, seasons, and targets with different magnitudes, relative metrics including the relative RMSE are additionally reported. Complementary indicators such as mean absolute error (MAE) and percentage-based errors provide robustness against outliers and improve interpretability. MAE is used to evaluate the models from Module 2, which is defined as follows:

$$mae = \left(\frac{1}{n}\right)\sum_{i=1}^{n} |y_i - x_i| \tag{3}$$

The second and third challenges are addressed with the use of prosumer clusters alongside the raw data streams as a means of normalization and validation of similar consumption and generation behaviors. For the second challenge of missing values and anomalies, the use of prosumer clusters mitigates the impact by enabling cluster-level aggregation and imputation. Specifically, when individual prosumer data are incomplete or contain anomalous measurements, values are imputed using cluster-level statistics, such as median or trimmed mean profiles, derived from time-aligned peers with similar consumption and generation characteristics. This approach preserves temporal structure while reducing sensitivity to outliers, as anomalous individual readings are addressed through aggregation within the cluster. For the third challenge, developing individual forecasts for each prosumer is not scalable when the microgrid expands and the number of prosumers increases. Therefore, prosumer clusters generated in Module 1 are used to identify disjoint segments and train a forecasting model for each profile and aggregate all profile forecasts to identify total consumption and generation. This module uses past 24 h consumption data as the input window. All the raw data stream values and continuous numerical features are scaled to [0, 1] using min-max normalization for the training process. All time-based features are converted to sin using $\sin\left(2\pi \cdot \frac{\text{hour}}{24}\right)$ and cos signals using $\cos\left(2\pi \cdot \frac{\text{hour}}{24}\right)$.

3.3. Module 3: microgrid auction for energy sharing

The final module, Module 3, conducts two functions: implementation of the internal auction and the optimization of the auction for energy sharing across prosumers. The module receives two inputs: prosumer clusters of selected granularity and the forecasts and forecast horizons for each prosumer and prosumer cluster. Prosumer clusters are used to identify the battery distribution in the microgrid, while forecasts and forecast horizons

are used as state parameters in the optimization of the internal microgrid auction, which is composed of two MARL methods.

3.3.1. Internal auction

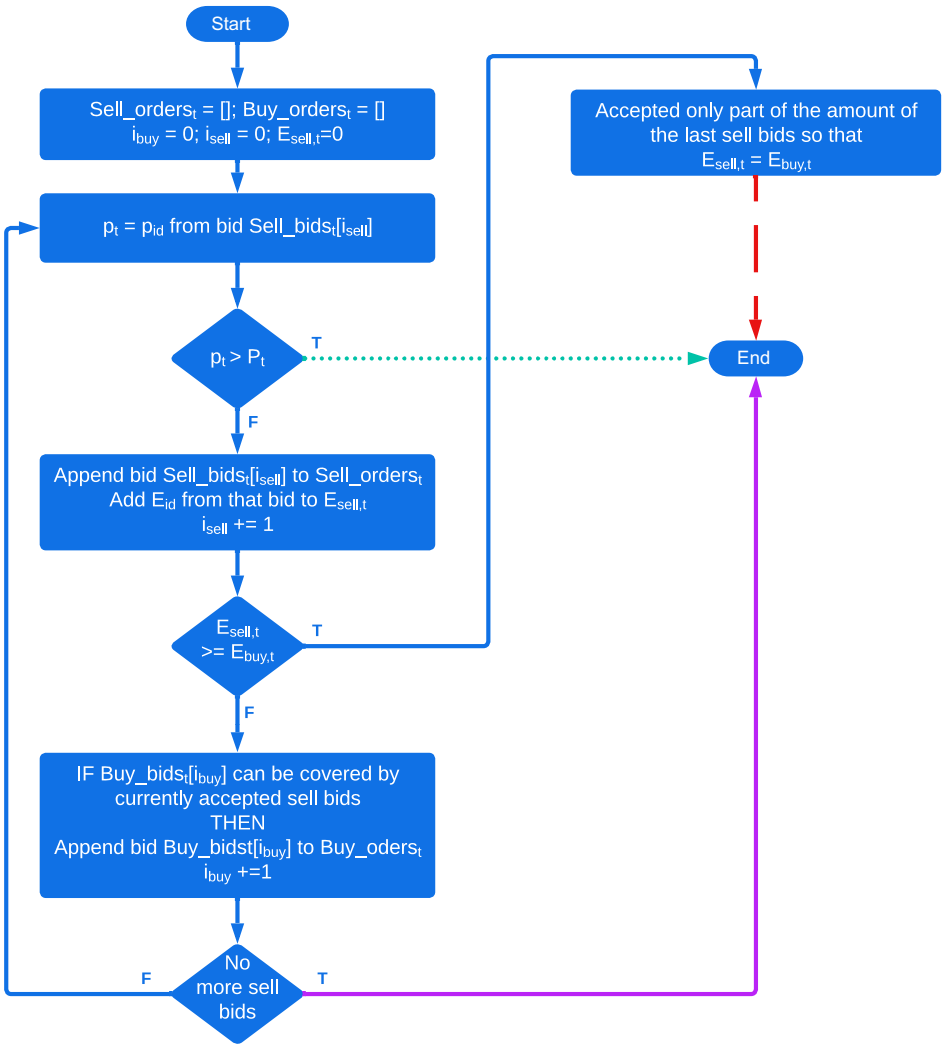
The internal auction function determines the energy-sharing structure of the prosumers, where each agent bids for buying or selling on the upcoming market epoch (the bid consists of the price at which the agent is willing to buy or sell as well as the amount of energy to be bought/sold in kWh). Thus, the action of the agent is to select the bid. The auction terminology is such that positive is to "sell," negative is to "buy," and accepted bids are called "orders." Figure 2 depicts a flowchart of the internal auction function. The auction runs hourly automatically for 24 h accepting bids from prosumer nodes. The proposed auction is an internal mechanism that does not have explicit separate auction steps like a normal auction and does not consider separate gate closure, operation, and settlement steps. Instead, bids should be evaluated prior to the top of each hour to determine the amount to buy/sell from/to the grid, covering the gate closure and settlement process. The proposed internal auction does not have trading prices. Instead, the auction is used to determine the accepted bids for each hour and the order of the accepted orders. The evaluation for hour t is specified in Figure 2. The evaluation begins with the cheapest sell bid. If it is cheaper than the grid price, it is accepted (i.e., it becomes an order), which is used to cover the cheapest buy bid, in part or fully: (1) if fully, that buy bid is accepted (i.e., it becomes an order) and the auction proceeds to the next buy bid, or, (2) if partly, the next sell bid is used to cover the buy bid that was only partially covered by the last sell bid. The auction then proceeds to evaluate the cheapest remaining sell bid. There are three ways in which the auction can end, each of which is indicated in the flowchart with a colored and formatted arrow as follows: (1) red and dashed: all of the buy bids were successfully covered by sell bids; (2) green and dotted: sell bids that had a price higher than the grid price were rejected, and the demand of the agents whose buy bids were not accepted in the auction were covered from the grid; and (3) violet and solid: all the sell bids were accepted, but they did not cover the demand from all of the agents, and the demand of the agents whose buy bids were not accepted in the auction were covered from the grid.

3.3.2. Energy-sharing strategy

The second function of this module is the optimization of the auction for energy sharing. It implements the distributed decision-making functionality for each prosumer node when participating in the auction. Every prosumer will be executing this function to

Figure 2

Unified Modeling Language (UML) diagram of the internal auction for energy-sharing structure of the prosumers (see Table 1 for terms)



determine the optimal action at each epoch of the auction. The action for each node is typically to either buy or sell into the microgrid, while the decision to buy or sell from the external grid is decided by the auction.

All prosumers have two objectives, where they need to minimize their own individual cost, and as a microgrid community, they need to minimize the overall cost of the microgrid. The potential conflicts of shared agent and independent agent approaches, which both have as an objective of minimizing cost, are addressed through the reward function by evaluating both community reward and individual reward and then selecting the most effective configuration. We utilize reinforcement learning to find optimal solutions that satisfy these two objectives. Single-agent reinforcement learning is unable to address the individual objectives of the participants due to the added complexities of nonstationarity, scalability, and partial observability [39]. Therefore, we have formulated internal auction actions of prosumers in the microgrid as a MARL optimization problem [40], which is solved in a distributed manner that supports scalable microgrid management while also accounting for individual prosumer objectives. The transformation of Single-Agent Reinforcement Learning (SARL) into MARL introduces the challenges of nonstationarity, scalability, and partial observability [39]. Nonstationarity is when the environment becomes nonstationary due to the actions of multiple agents [41]; scalability of the joint action space, which increases with the number of agents [42]; and partial observability, where different agents observe different instances of state [43]. Three types of MARL approaches attempt to address these three challenges: independent agents, joint action space, and centralized training of decentralized policies. Independent agents are the most widely used [44] and straightforward as they ignore the nonstationarity condition and implement each agent as a separate SARL agent [45]. In joint action spaces, an SARL agent represents the multi-agent environment using joint or shared actions. This method resolves the nonstationarity issue by representing the multi-agent environment using a single agent [46]. But this method has issues with scalability in a multi-agent environment, as it cannot take advantage of multiple agents and needs to train a single larger agent. The third approach is to use decentralized policies similar to independent agents but use a centralized training process [44]. Further, this can support partial observability through local states and actions in each agent [39], as reported in several recent approaches [47, 48].

Based on recent literature, Multi-Agent Proximal Policy Optimization (MAPPO)-based independent agent algorithms are able to mitigate the nonstationarity of the environment using techniques from proximal policy optimization (PPO), such as policy clipping [49]. In this work, we use independent agent MAPPO and centralized training approaches in MARL to optimize energy sharing in a prosumer microgrid. The proposed MAPPO algorithm follows the architecture suggested in Reference [49] and learns decentralized independent policies for each agent based on individual policy clipping.

3.3.3. *Optimization function*

The optimization function aims to maximize energy sharing and minimize the emission and cost of the microgrid. The cost is quantified by the number of units purchased by the grid multiplied by the spot price at that time P_t. To achieve this objective, the function of an individual microgrid entity can be defined as a cost for an individual epoch, considering the three energy generation and consumption units: each microgrid entity's energy consumption P_i^{load}, each microgrid entity's PV energy generation P_i^{PV}, and battery energy storage change P_i^{BES}. The identified objective function is shown in Equation (4).

$$\triangle J_i(t) = P_t\left(P_i^{load}(t) - P_i^{PV}(t) + P_i^{BES}(t)\right) \tag{4}$$

The energy storage system of each prosumer has a constraint that is required to protect the energy storage from high throughput. This restriction ensures the minimum and maximum state of energy (SOE) of the battery. This is important for the safety of the batteries [50, 51]. Equation (5) shows the constraint of the SOE value, and Equation (6) shows the calculation of the SOE value.

$$SOE_{i,min} \le SOE_i \le SOE_{i,max} \tag{5}$$

$$SOE_i(t) = SOE_i(t-1) + \frac{\eta P_i^{BES}}{BES_i^{max}} \tag{6}$$

Here, P_i^{BES} is the discharge/charge amount for the building i, η is the charging/discharging efficiency coefficient, and BES_i^{max} is the battery capacity.

The total cost of the microgrid can be identified as J. The purpose of the multi-agent system is to minimize this objective function. Equation (7) shows the objective function of the system. Here, M is the total number of epochs considered for optimization, and N are all individual participants in the microgrid.

$$J = \sum_{t=1}^{M}\sum_{i=1}^{N}\triangle J_i(t) \tag{7}$$

This generalized energy system can be modeled as an optimization of two objectives: first is to consider the individual cost of each prosumer, and second is to use the overall cost of the microgrid by aggregating the individual cost over all participants. In this research, both objective functions were empirically evaluated, and the second was selected due to improved value. This is expected, as the objective of the multi-agent system is to reduce the overall external cost of the microgrid. The cost function to consider for minimization is the total cost(J) of all the epochs.

3.3.4. *The MARL approach*

This section presents the MARL approach for the identified optimization function. Reinforcement optimization is formulated by considering a Markov decision process (MDP) and defined as a tuple of (S, A, P, R, γ). Here, S is the environment state space, A is the action space, P is the transition probabilities, R is the reward, and γ is the discount factor. When there are multiple agents, the MDP process is no longer valid, and actions from individual agents affect the other agent dynamics. The extension of MDP, known as a Markov game, is used to formalize the multi-agent environment. The MARL parameter tuple will change as $(N, S, \{A\}^{i\in N}, P, \{R\}^{i\in N}, \gamma)$. In this tuple, N is the number of agents ($N > 1$), S is the state space, R^i is the reward function, and γ is the discount factor, $\gamma \in [0, 1]$. A^i is the action space for the i th agent, and the joint action space can be identified as $A = A^1 \times A^2 ... \times A^N$. P is the transition probability $P : S \times A \to \delta(S)$ for the state $s' \in S$ when given starting state $s \in S$ and joint action $a \in A$.

$$V_i^{\pi}(s) = \sum_{a\in A}\pi(s,a)\sum P(s,a,s')\left[R^i(s,a,s') + \gamma V_i(s')\right] \tag{8}$$

Joint policy of multi-agents can be identified as $\pi(s,a) = \prod_j \pi_j(s,a_j)$, and it is determined by the value function in

Equation (8). Here, a is the joint action that can be demonstrated using the common notation for opponent

set – i = N

i as $a = (a_i, a_{-i})$.

$$\pi_i^* (s, a, \pi_{-i}) = argmax\, V_i^{(\pi_i, \pi_{-i})}(s) \tag{9}$$

Optimal policy for agent *i* can be identified as Equation (9), and it depends on the policies of the remaining agents. In a MARL setting, multiple agents can interact with the environment and update the state of the environment; these agents can collaborate or compete in a single environment, and individual agents can have the same or multiple policies [52]. Moreover, the state is determined by the environment state and agents' own state, and the reward value of a single agent is determined not only by their own action but also by the other agent's action. Due to these complexities, there can be a scenario where the optimization problem does not converge, and the methods to use for a multi-agent scenario are dependent on the use case. Therefore, energy sharing of the microgrid community environment is a typical scenario for MARL optimization, where each prosumer is considered a separate agent with their own goals and determines their own actions to participate in the internal auction. MARL approach provides a scalable solution considering the individual goals of the agents and addresses the added complexity that arises from the growth of the dimensions of the input state [50].

Action space: The individual agent determines the bids for the internal auction, which includes the bidding price, bidding volume (amount), and whether the bid is a sell or buy order. To achieve this, we use two separate values for price and amount. The sign of the price value will determine if the bid is a buy or sell order: a negative price means a sell order, and a positive price means a buy order. To simplify the MARL agents, these two values are discretized as follows: the amount is kept in a constant value (CA), which is based on the previous year's max amount, and the price is considered as two discrete values for buy and sell (α), where $\alpha \in (-1, 1)$. When considering the bid, the auction price is converted to a continuous value using a linear scaling of the current grid price. The total number of such tuples within the action space is determined by the number of agents and the number of participant buildings in the microgrid.

State space: The multi-agent environment state space has two components: the agent's own state space and the environment space. The agent's own states are visible to agents, but environment space visibility is determined by the agent's visibility of other agents. This is identified as full observability or partial observability. Table 3 shows the complete state space used for the MARL solution assuming full observability. For the state transition of the agent space from the action, Equation (10) determines the discharge or charge amount.

$$P_i^{load(t)} + \alpha \cdot CV = P_{BES} + P_i^{PV(t)} \tag{10}$$

Reward function: Determining the reward function is the most important aspect of an reinforcement learning (RL) solution. In this work, we evaluated three reward functions to identify the optimum solution for the MARL optimization: (1) common cost of the microgrid $\sum_{i=1}^{N} \triangle J_i(t)$ at each epoch, (2) individual cost of the building (Equation (3)), and (3) reward function, which evaluates accepted buy order as positive value and sell order as negative value, multiply by current price (P_t). The reward function is selected empirically by evaluating the results for the four seasons. In this experiment, the first reward function identified is used with the independent agent method.

4. Experiments

The proposed AI framework and its constituent modules were empirically evaluated using a real-world case study based in a tertiary education setting in South East Australia. The case study consisted of data for energy consumption and renewables generation from 56 buildings for two years, Jan 1, 2023, to Jan 1, 2025. Both consumption and generation data are streamed into a centralized server in 15 min intervals. The 56 buildings are used for various purposes, such as lecture theaters and classrooms for teaching, administration offices, student accommodation, sport and recreation, and retail. Battery storage is available for 28 buildings. Each building will work as an edge node, predict the generation and consumption for the next epoch, and determine the bidding action to participate in the internal auction. In this scenario, we assume the full observability between agents as the community is a university and individual building information is shared among others using the central agent.

Table 3
Summary of the state space values with each category

Category	Value
Environment state	Microgrid total load ($\sum_{i=1}^{N} P_i^{load}(t)$)
	Microgrid total PV generation ($\sum_{i=1}^{N} P_i^{PV}(t)$)
	Current market price
	Predicted microgrid load at $t+1$, $\sum_{i=1}^{N} \tilde{P}_i^{load}(t+1)$
	Microgrid total PV generation at $t+1$, ($\sum_{i=1}^{N} \tilde{P}_i^{PV}(t+1)$)
	Month
	The day of the week
	Hour of the day
Agent state	Agent load ($P_i^{load}(t)$)
	Agent PV generation ($P_i^{PV}(t)$)
	Current SOE value

Table 4
Comparison of five state-of-the-art MARL algorithms across eight capabilities relevant to microgrid energy-sharing auctions

Capabilities	QMIX	Independent Q-Learning (IQL)	MADDPG	MAAC	MAPPO
Action space	Discrete	Discrete/Cont	Continuous	Discrete/Cont	Discrete/Cont
Training methods	Centralized Critic, Decentralized Actor	Independent Learning	Centralized Critic, Decentralized Actor	Centralized Critic, Decentralized Actor	Centralized Critic, Decentralized Actor
Optimization method	Value Decomposition	Q-learning	DDPG-based (Deterministic Policy Gradient)	Actor-Critic with Q-function Decomposition	PPO-based (Policy Gradient)
Stability	High (in cooperative settings)	Low (due to nonstationarity)	Moderate (requires tuning)	High (due to Q-function decomposition)	High (due to PPO stability)
Sample efficiency	High	Low	Moderate	High	High
Scalability	High (for cooperative tasks)	Excellent (independent agents)	Moderate (centralized critic)	Moderate (centralized critic)	High
Handling coordination	High	Low (independent learning)	Moderate	High	High
Multi-agent settings	Cooperative settings	Simple environments with independent agents	Cooperative or mixed environments	Cooperative tasks, both discrete and continuous	Complex multi-agent environments (cooperative/competitive)

Due to the diverse capabilities of MARL algorithms, it was pertinent to compare the state of the art across numerous attributes for the selection of a suitable algorithm for the proposed framework and the nontrivial energy-sharing microgrid auctions environment. For instance, Multi-Agent Deep Deterministic Policy Gradient (MADDPG) performs well in continuous action spaces, Multi-Agent Proximal Policy Optimization (MAPPO) provides improved stability due to its use of PPO, and Multi-Agent Actor-Critic (MAAC) performs well in scenarios requiring tight coordination among agents. Based on this comparison (Table 4), MAPPO was selected as the most suitable algorithm due to the capacity for high optimization, stability, coordination, diversity of multi-agent settings, and Centralized Critic, Decentralized Actor training method in continuous spaces.

The agents were trained using simulated online MARL over historical market data from 2023, rather than a static batch learning procedure. Although the data are historical, the training process explicitly preserves the temporal dynamics and strategic interactions among agents. The 2023 dataset was used to construct a sequential market simulation environment, where each trading interval (e.g., hourly/day-ahead auction) represents one environment step. At each step, all agents simultaneously observe their local states (including price signals, demand/generation forecasts when available, and historical clearing outcomes) and submit bidding actions to the simulated auction mechanism. Agents are trained online within each episode, so that policy updates occur iteratively as agents interact with one another and the environment over the full 2023 timeline. This enables agents to adapt to nonstationarity induced by other learning agents. Multiple episodes over the 2023 period are executed to ensure convergence and policy stability. For evaluation, the learned policies are frozen and deployed in an out-of-sample simulation using 2024 data, without further learning or parameter updates. This temporal separation prevents information leakage and allows performance to be assessed under dynamic conditions. This training–evaluation split reflects a realistic deployment setting in which bidding agents are trained on historical market behavior and then executed in future markets without retraining.

The independent agent method uses a separate policy for each agent, and the shared agent method shares the same policy for all the agents. The MARL hyperparameters were optimized using three methods: Grid Search, Bayesian, and Gradient-based. Grid Search was guided by cross-validation on the training set, Bayesian guided by a convergence objective evaluated on the validation dataset, and Gradient-based was guided by a hypernetwork trained to approximate the best response for multi-agent functions. The resulting hyperparameters were then used in the experiments for the PPO algorithm, as identified in Table 5.

Table 5
Selected PPO hyperparameters

Hyperparameter	Value
γ	0.99
Batch size (horizon)	2500
Learning rate (Adam step size)	0.001
GAE parameter (lambda)	1
Clipping parameter	0.3

All experiments were conducted using a discrete-time simulation environment developed in Python v3.9. The MARL framework and forecasting models were implemented using PyTorch v1.13, NumPy, and Pandas. The energy-sharing auctions, including prosumer consumption, generation, and battery constraints, were simulated within a custom-built environment executed at each decision step. Experiments were run on a workstation equipped with an Intel Core i7 CPU, 32 GB RAM, and an NVIDIA RTX 3080 GPU with 10 GB memory. All training

and evaluation were performed on a single machine without distributed computing. This hardware configuration was sufficient to support the computational requirements of the proposed MARL framework and ensures reproducibility of the reported results.

4.1. Experiment 1

This experiment evaluates a suitable MARL method for the selected dataset from the two methods available in the AI framework: independent agents and shared agents' methods. To empirically evaluate the two methods, both MARL agents are trained on the 2023 data and evaluated using the 2024 data. In Figure 3, we compare the MARL training process using two methods. The graph shows the mean reward plotted against the training iteration. Up to around 100 epochs, the reward increases exponentially and then saturates, indicating the convergence of the reinforcement learning process. However, after around 700 episodes, the independent agent method passes the shared parameter method and achieves an overall higher mean reward. This demonstrates that compared to single RL agent models, the multi-agent models are unstable, and that the overall highest mean reward is higher in the independent agent method. Figure 4 presents the accumulated error for the two MARL methods, where the independent agent method has a lower cost compared to the shared agent method.

4.2. Experiment 2

This experiment evaluates the time-series forecasting ensemble module in the MARL energy-sharing strategy. Given the seasonality of forecasts, this experiment takes into account the four seasons of Australia. Table 6 shows the selected date ranges for the experiments from the four seasons. In this table, the first column, "Season," indicates the season in which the data are selected; the second column, "Training Date," signifies the selected starting date in the training dataset for each season; and the third column, "Testing Date," identifies the starting date in the testing dataset for each season. In this experiment, we selected a subset of five prosumers with variable and dynamic consumption and generation patterns.

Figure 3
Comparison of the two MARL methods: independent agent and shared parameter method

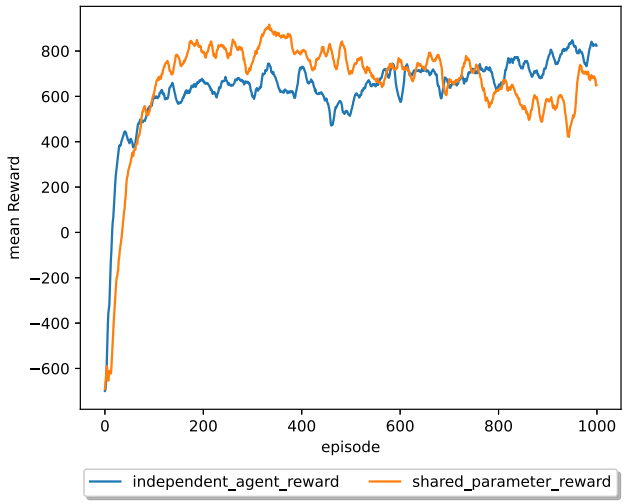


Figure 4
Accumulated error of the two MARL methods: independent agent and shared parameter method

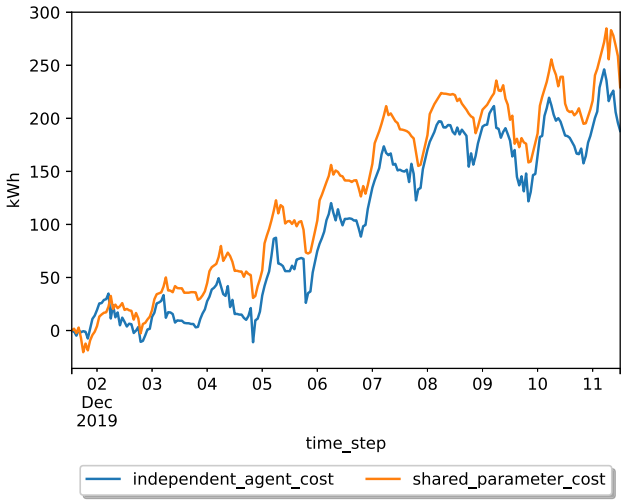


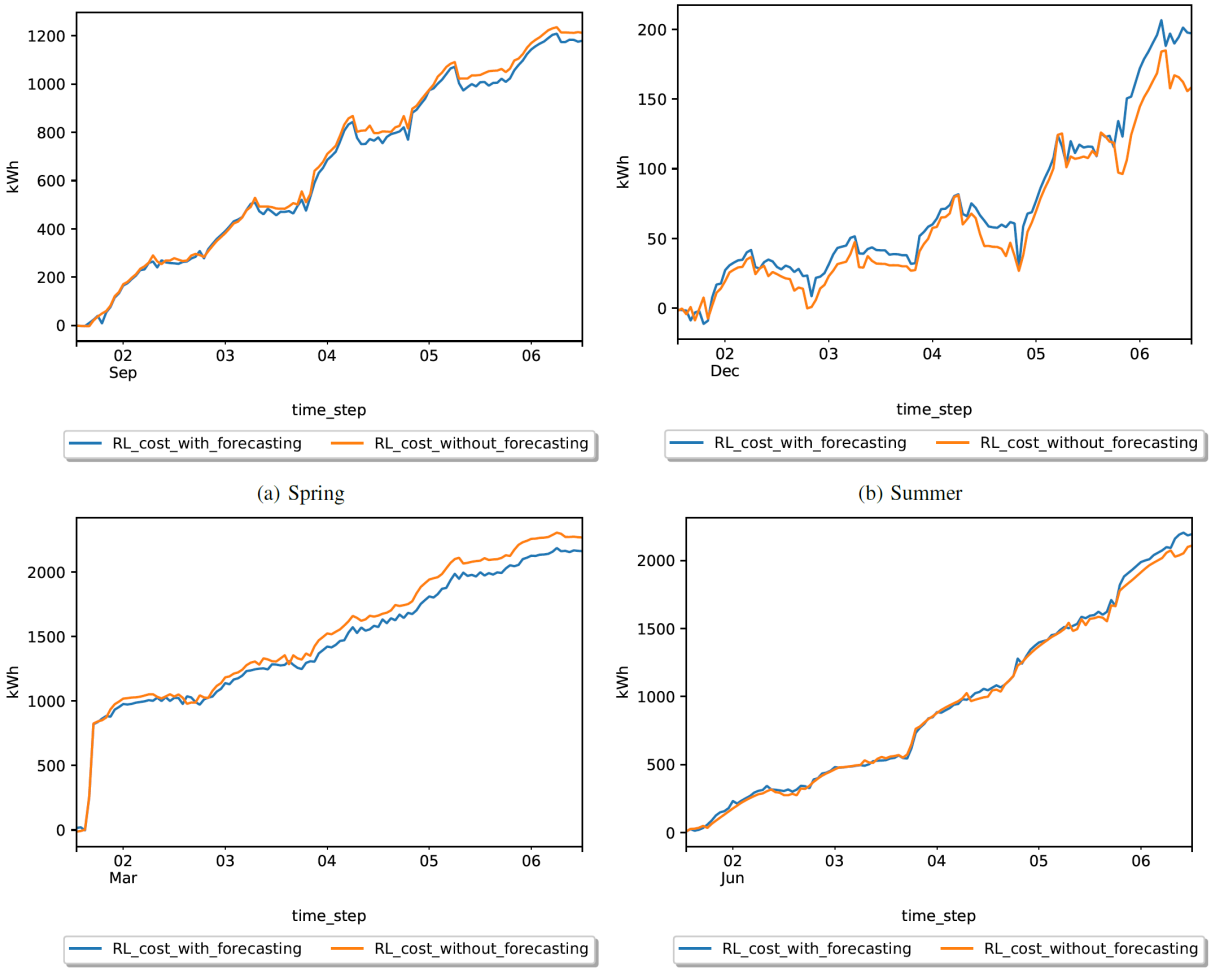
Table 6
Training and testing periods for multi-agent reinforcement learning for four seasons

Season	Training date	Testing date
Spring	01-09-2023	02-09-2024
Summer	01-12-2023	01-12-2024
Autumn	01-03-2023	01-03-2024
Winter	01-06-2023	01-06-2024

The experiment uses independent agents as the implementation method of MARL. All multi-agents are trained on the data from 2023 and evaluated using the 2024 data (date ranges as shown in Table 6). To evaluate the forecasting method, multi-agents are trained in two approaches, with and without the forecasting data. The MARL method determines bidding to participate in the auction and calculates the accumulated cost for the first five-day period starting from the dates presented in Table 6. The forecasting method to utilize for the experiment is empirically selected from the available forecasting methods in the AI framework.

Figure 5 shows for each season how MARL bidding in the auction works with and without forecasting data of consumption and generation. According to this comparison of two seasons with forecasting information, MARL bidding action prediction generates a lower cost. Both autumn and spring bidding action with forecasting information generates lower accumulated costs compared to bidding without forecasting. This implies that high variances in spring and autumn are detected in the forecasts, and this enables the MARL bidding action to predict more accurately. In contrast, summer and winter bidding actions produce lower costs without the forecasting information. The contrasting behavior observed in summer and winter is likely due to the relative stability of consumption and generation patterns during these seasons. In summer, solar generation follows a more predictable diurnal profile with lower short-term variability, while in winter, consumption patterns tend to be dominated by stable demand. Under such conditions, the additional forecasting information provides a limited marginal benefit with some

Figure 5
Comparison of MARL accumulated costing with and without forecasting



(a) Spring (b) Summer

(c) Autumn (d) Winter

prediction errors that propagate into the bidding policy. As a result, MARL agents relying solely on recent observations can achieve comparable or lower costs than those incorporating forecasts. In the next experiments, these findings are used to determine the training of the MARL agents for each season.

4.3. Experiment 3

This experiment evaluates the proposed independent agent MARL method with three heuristic methods that also utilize the same generalized optimization function. For this comparison, heuristics are more suitable than MPC due to its limitations in applicability to peer-to-peer energy-sharing auctions. First, MPC typically relies on accurate and explicit system models, including demand dynamics and generation profiles. In auction-based energy-sharing settings, prices and bidding outcomes are non-deterministic and emerge endogenously through interactions among multiple agents, limiting the validity of such model assumptions. Second, standard MPC formulations do not explicitly account for strategic behavior, as they generally assume either centralized coordination or price-taking agents. This means MPC is unable to capture prosumers dynamically adapting their bids in response to competitors' actions. Third, extending MPC to multi-agent auction environments would require either centralized optimization with full information sharing or iterative coordination schemes, both of which are impractical in peer-to-peer markets due to scalability, privacy, and communication constraints. In contrast, heuristic bidding strategies reflect the types of decision rules commonly adopted by prosumers in practical deployments, requiring minimal information, no explicit system models, and low computational overhead. This aligns closely with the proposed MARL framework, which directly learns decentralized bidding policies from interaction data, enabling agents to adapt to nonstationary auction environments without relying on explicit system models.

The independent agent MARL method utilizes battery allocation from Experiment 1 and time-series forecasts from Experiment 2. The MARL agents are trained on 2023 data for the four seasons, and for each season, a separate MARL model is identified. The three heuristic methods used in this experiment are as follows.

Heuristic Method 1: The first heuristic method uses the minimalistic solution, which will bid to the auction based on actual consumption and generation values. This method will produce a sell bid when there is excess energy from generation. Price and amount are determined by the previous external price and the actual excess amount. If there is a lower generation than

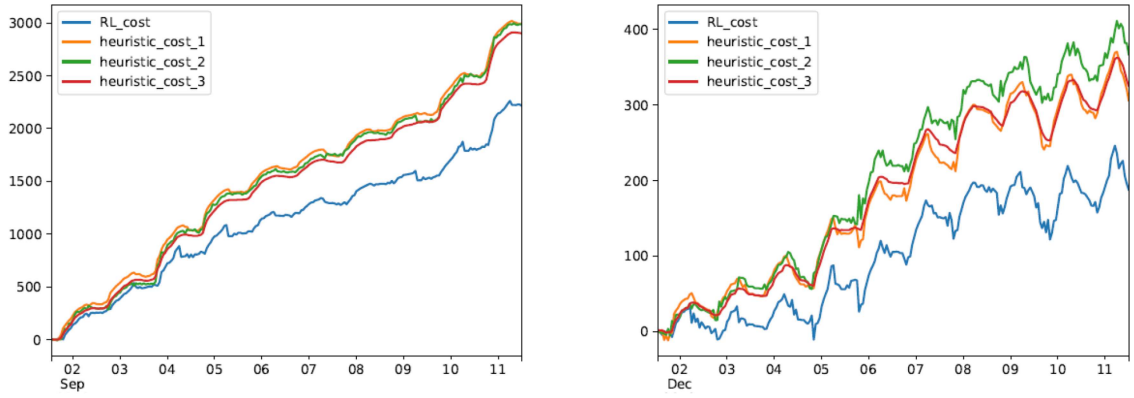
consumption, it produces a buy bid for the auction for the previous external price.

Heuristic Method 2: The second heuristic method uses random values to determine the amount, and the amount values are randomly picked from a normal distribution constraint to the maximum and minimum identified. Price is determined by the previous external price similar to heuristic method 1.

Heuristic Method 3: The third method uses the forecast to determine buy and sell orders. If the predicted price is less than the predefined threshold value, then the order is set to "buy order"; otherwise, it is set to "sell order." This will ensure buying from the grid when the price is less at the predicted price. The amount is determined by the random value similar to heuristic method 2, and the threshold value is determined by the previous year's average external price. Figure 6 provides a cost comparison between the three heuristic methods and the MARL method for two seasons, spring and summer. Accordingly, MARL is the lowest of all four methods, and the cost reduction is tenfold lower in summer due to the lower consumption and higher generation values.

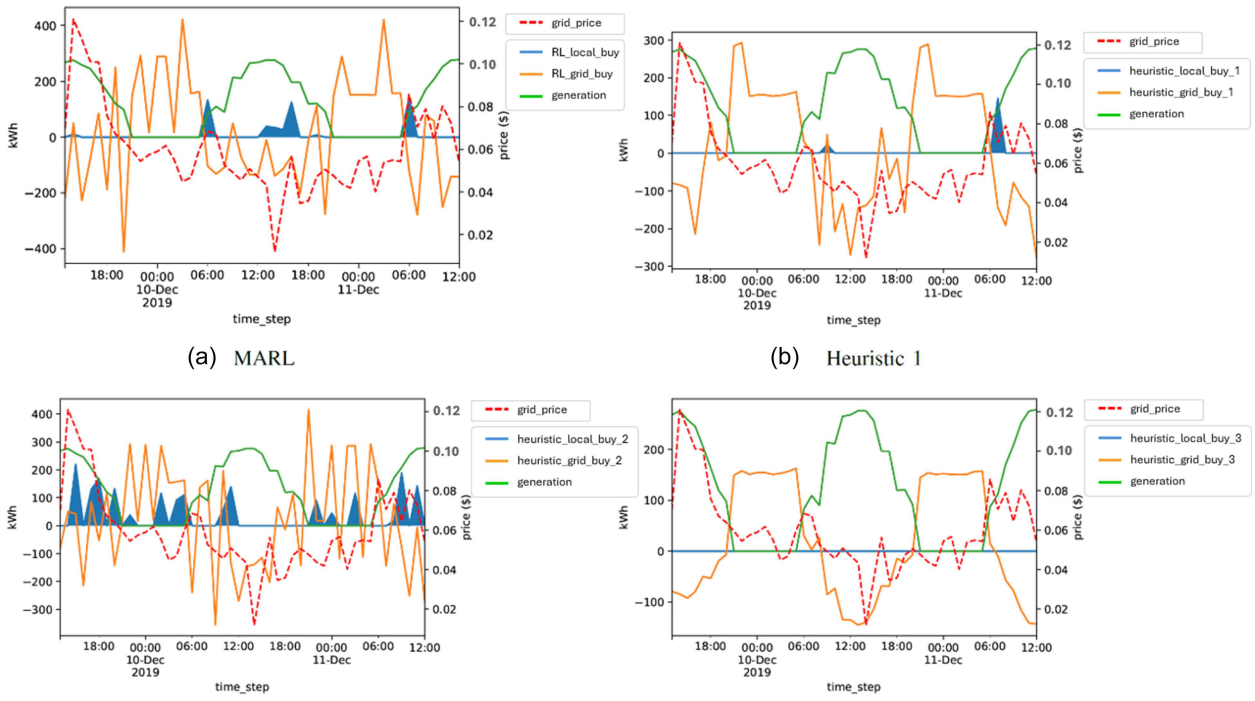
We analyzed this gain in the MARL method by comparing the last 24 time steps of summer with the local buy values between the individual participants of the microgrid. Figure 7 shows the comparison between the MARL method and three heuristic methods. According to these graphs, the MARL method learns to increase the local grid buy values when the external

Figure 6
Cost comparison of the heuristic methods and independent agent MARL method for spring and summer



(a) Spring (b) Summer

Figure 7
Comparison of price and generation by local and grid buy for the MARL method and three heuristics



(a) MARL (b) Heuristic 1

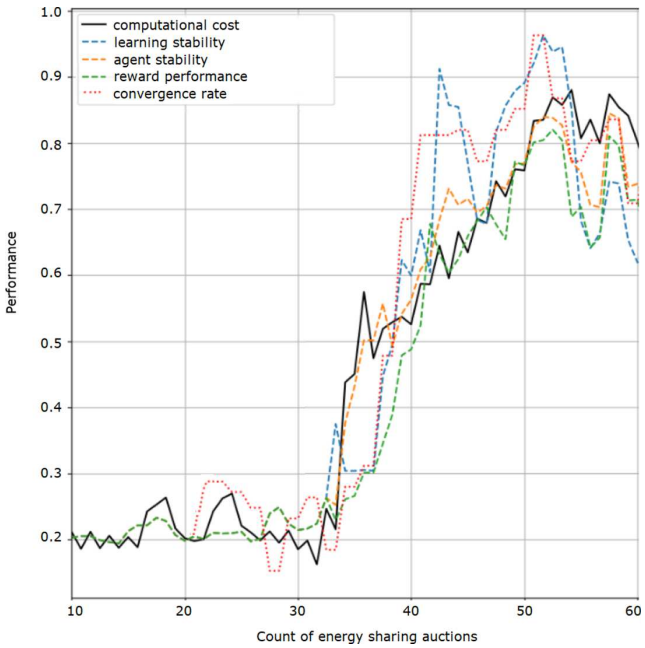
(c) Heuristic 2 (d) Heuristic 3

energy price is high compared to the three heuristic methods. This results in lower external energy buy in high price points and high external buys when lower price points. For example, on both December 10 and December 11 at 06:00, there are high price points where the local buy is high (blue shaded regions). Furthermore, at approximately 16:00 on December 10, another instance of high local buying activity is observed. All high local buy points correspond to high grid price points. On the other hand, heuristic method 1 has only one high local buy point on December 11 at 07:00 when the price is at its lowest, and heuristic method 2 has considerable local buy points (blue shaded regions) randomly in both high and low price points, and in the considered 24 time steps, heuristic method 3 has no high local buy points. Here, local buy means local energy sharing between the nodes of the microgrid. Local buy amount is calculated by iterating all the sell orders and aggregating sell order amounts that have a local buying entity, which is a microgrid node.

4.4. Experiment 4

Experiment 4 evaluates the scalability of the proposed framework for an increasing number of auctions in more complex environments with increased consumption, generation, and provision. As the number of auctions increases, the complexity of decision-making, coordination among agents, and the overall training process becomes nontrivial. This scalability assessment can determine how effectively the framework adapts to larger-scale problems without compromising performance or stability. Scalability testing further enables identifying potential bottlenecks in computational resources, learning time, and reward convergence. As depicted in Figure 8, we have evaluated the framework for 60 auctions and presented metrics for cost, stability, reward, and convergence. Overall, the increasing number of auctions leads to an increasing computational cost; however, the multi-agent informational flows of the framework ensure that learning stability, agent stability, and reward function are also increasing in performance. This balance between the cost of learning and efficiency of increasing auctions is indicative of related work on MAPPO reporting slightly higher rewards for fewer auctions and stable performance with more auctions.

Figure 8
Scalability of the proposed framework for increasing number of auctions



5. Conclusion

In this paper, we have formulated the prosumer microgrid as independent, self-serving agents who are motivated to optimize their own financial goals, and by completing their own goals, they are able to contribute toward community financial goals. An internal auction setting enables the competitive behaviors required to achieve individual and community goals. We leveraged MARL to optimize each agent's goals, as it enables scalability and computational efficiency by allowing distributed processing of individual agents. However, it is challenging to converge to an optimized solution due to the high complexity that arises from the dynamic environment of an individual self-serving agent because the environment itself is defined by other self-serving agents. We proposed two MARL methodologies—independent agent and shared agent—to overcome this challenge. Moreover, to improve the optimization process, two interdependent modules were introduced to identify clusters of prosumer agents and forecast their consumption and generation. The clustering module identifies clusters of self-interested agents, which is utilized in identifying the optimum energy storage distribution in agents and as a scalability enabling technique in the forecasting module. The forecasting module supports MARL by predicting the uncertainties that arise from factors external to the microgrid. This AI framework, consisting of a structure-adapting unsupervised learning for prosumer clustering, a time-series forecasting ensemble for forecasting, and a continuous internal auction with a MARL optimization, is demonstrated on the real-world microgrid setting of a large multi-campus tertiary education institution. Due to the complex configuration of a prosumer energy-sharing microgrid setting, we have recognized further evaluation of the proposed framework and its learning capabilities as future work that is outside the scope of this article. This includes evaluation of the framework in diverse microgrid environments such as commercial, residential, and industrial scenarios, as well as a comparison with other similar methods that leverage prosumer clustering and time-series forecasting to inform and enable a continuous internal auction using MARL capabilities. Further studies will also consider partial observability, fairness-aware objectives, and transferability of learned policies across seasons and microgrid environments, as well as real-time deployment constraints and regulatory considerations.

Funding Support

The work of authors (DDS, TK, PR, NM, AJ) affiliated with the La Trobe Artificial Intelligence Institute was supported by a grant from the Department of Climate Change, Energy, the Environment and Water of the Australian Federal Government, as part of the International Clean Innovation Researcher Networks (ICIRN) program, grant number ICIRN000077.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

The data that support the findings of this study are openly available at https://github.com/CDAC-lab/UNICON.

Author Contribution Statement

Daswin De Silva: Conceptualization, Methodology, Formal analysis, Writing – original draft, Supervision, Funding acquisition. **Thimal Kempitiya:** Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft. **Nuwan Madhusanka:** Software, Validation, Resources, Data curation, Writing – original draft. **Prabod Rathnayaka:** Formal analysis, Investigation, Resources, Writing – original draft, Visualization. **Nishan Mills:** Methodology, Validation, Investigation, Writing – original draft, Supervision. **Andrew Jennings:** Conceptualization, Formal analysis, Resources, Writing – original draft, Supervision. **Milos Manic:** Conceptualization, Methodology, Validation, Writing – original draft, Supervision.

References

[1] Trivedi, R., & Khadem, S. (2022). Implementation of artificial intelligence techniques in microgrid control environment: Current progress and future scopes. *Energy and AI*, *8*, 100147. https://doi.org/10.1016/j.egyai.2022.100147

[2] Uddin, M., Mo, H., Dong, D., Elsawah, S., Zhu, J., & Guerrero, J. M. (2023). Microgrids: A review, outstanding issues and future trends. *Energy Strategy Reviews*, *49*, 101127. https://doi.org/10.1016/j.esr.2023.101127

[3] Alam, M. S., Hossain, M. A., Shafiullah, M., Islam, A., Choudhury, M. S. H., Faruque, M. O., & Abido, M. A. (2024). Renewable energy integration with DC microgrids: Challenges and opportunities. *Electric Power Systems Research*, *234*, 110548. https://doi.org/10.1016/j.epsr.2024.110548

[4] Ahmad, S., Shafiullah, M., Ahmed, C. B., & Alowaifeer, M. (2023). A review of microgrid energy management and control strategies. *IEEE Access*, *11*, 21729–21757. https://doi.org/10.1109/ACCESS.2023.3248511

[5] Pavão, A., Marot, A., Sintes, J., Möllerstedt, V. E., Crochepierre, L., Chaouache, K., . . . , & Guyon, I. (2025). AI challenge for safe and low carbon power grid operation. *Energy and AI*, *22*, 100564. https://doi.org/10.1016/j.egyai.2025.100564

[6] Czétány, L., Vámos, V., Horváth, M., Szalay, Z., Mota-Babiloni, A., Deme-Bélafi, Z., & Csoknyai, T. (2021). Development of electricity consumption profiles of residential buildings based on smart meter data clustering. *Energy and Buildings*, *252*, 111376. https://doi.org/10.1016/j.enbuild.2021.111376

[7] Zhan, S., & Chong, A. (2021). Building occupancy and energy consumption: Case studies across building types. *Energy and Built Environment*, *2*(2), 167–174. https://doi.org/10.1016/j.enbenv.2020.08.001

[8] Peter, N., Gupta, P., & Goel, N. (2025). Intelligent strategies for microgrid protection: A comprehensive review. *Applied Energy*, *379*, 124901. https://doi.org/10.1016/j.apenergy.2024.124901

[9] Takiddin, A., Ismail, M., Zafar, U., & Serpedin, E. (2022). Deep autoencoder-based anomaly detection of electricity theft cyberattacks in smart grids. *IEEE Systems Journal*, *16*(3), 4106–4117. https://doi.org/10.1109/JSYST.2021.3136683

[10] Tabassum, T., Toker, O., & Khalghani, M. R. (2024). Cyber–physical anomaly detection for inverter-based microgrid using autoencoder neural network. *Applied Energy*, *355*, 122283. https://doi.org/10.1016/j.apenergy.2023.122283

[11] Jin, B., & Xu, X. (2024). Forecasts of China mainland new energy index prices through Gaussian process regressions. *Journal of Clean Energy and Energy Storage*, *1*, 2450006. https://doi.org/10.1142/S2811034X24500060

[12] Lemos-Vinasco, J., Bacher, P., & Møller, J. K. (2021). Probabilistic load forecasting considering temporal correlation: Online models for the prediction of households' electrical load. *Applied Energy*, *303*, 117594. https://doi.org/10.1016/j.apenergy.2021.117594

[13] Serrano-Guerrero, X., Briceño-León, M., Clairand, J.-M., & Escrivá-Escrivá, G. (2021). A new interval prediction methodology for short-term electric load forecasting based on pattern recognition. *Applied Energy*, *297*, 117173. https://doi.org/10.1016/j.apenergy.2021.117173

[14] Xu, X., & Zhang, Y. (2023). An integrated vector error correction and directed acyclic graph method for investigating contemporaneous causalities. *Decision Analytics Journal*, *7*, 100229. https://doi.org/10.1016/j.dajour.2023.100229

[15] Entezari, A., Aslani, A., Zahedi, R., & Noorollahi, Y. (2023). Artificial intelligence and machine learning in energy systems: A bibliographic perspective. *Energy Strategy Reviews*, *45*, 101017. https://doi.org/10.1016/j.esr.2022.101017

[16] Liu, L.-N., Yang, G.-H., & Wasly, S. (2024). Distributed predefined-time dual-mode energy management for a microgrid over event-triggered communication. *IEEE Transactions on Industrial Informatics*, *20*(3), 3295–3305. https://doi.org/10.1109/TII.2023.3304025

[17] Zhang, Q., Mu, Y., Jia, H., Yu, X., & Hou, K. (2024). Distributionally robust optimization configuration method for island microgrid considering extreme scenarios. *Energy and AI*, *17*, 100389. https://doi.org/10.1016/j.egyai.2024.100389

[18] Ahsan, S. M., Gholizadeh, N., & Musilek, P. (2025). Multi-agent systems in networked microgrids: Reinforcement learning and strategic pricing mechanisms. *Renewable Energy*, *254*, 123678. https://doi.org/10.1016/j.renene.2025.123678

[19] Shuai, H., Li, F., Pulgar-Painemal, H., & Xue, Y. (2021). Branching dueling Q-network-based online scheduling of a microgrid with distributed energy storage systems. *IEEE Transactions on Smart Grid*, *12*(6), 5479–5482. https://doi.org/10.1109/TSG.2021.3103405

[20] Guo, C., Wang, X., Zheng, Y., & Zhang, F. (2021). Optimal energy management of multi-microgrids connected to distribution system based on deep reinforcement learning. *International Journal of Electrical Power & Energy Systems*, *131*, 107048. https://doi.org/10.1016/j.ijepes.2021.107048

[21] Sierla, S., Pourakbari-Kasmaei, M., & Vyatkin, V. (2022). A taxonomy of machine learning applications for virtual power plants and home/building energy management systems. *Automation in Construction*, *136*, 104174. https://doi.org/10.1016/j.autcon.2022.104174

[22] Sumanasena, V., Gunasekara, L., Kahawala, S., Mills, N., de Silva, D., Jalili, M., . . . , & Jennings, A. (2023). Artificial intelligence for electric vehicle infrastructure: Demand profiling, data augmentation, demand forecasting, demand explainability and charge optimisation. *Energies*, *16*(5), 2245. https://doi.org/10.3390/en16052245

[23] Li, S., Zhao, P., Gu, C., Li, J., Cheng, S., & Xu, M. (2023). Battery protective electric vehicle charging

management in renewable energy system. *IEEE Transactions on Industrial Informatics*, *19*(2), 1312–1321. https://doi.org/10.1109/TII.2022.3184398

[24] Utkarsh, K., & Ding, F. (2022). Self-organizing map-based resilience quantification and resilient control of distribution systems under extreme events. *IEEE Transactions on Smart Grid*, *13*(3), 1923–1937. https://doi.org/10.1109/TSG.2022.3150226

[25] de Silva, D., Yu, X., Alahakoon, D., & Holmes, G. (2011). Semi-supervised classification of characterized patterns for demand forecasting using smart meters. In *2011 International Conference on Electrical Machines and Systems*, 1–6. https://doi.org/10.1109/ICEMS.2011.6073434

[26] Lee, S., & Choi, D.-H. (2020). Energy management of smart home with home appliances, energy storage system and electric vehicle: A hierarchical deep reinforcement learning approach. *Sensors*, *20*(7), 2157. https://doi.org/10.3390/s20072157

[27] Liao, Z., & Coimbra, C. F. M. (2024). Hybrid solar irradiance nowcasting and forecasting with the SCOPE method and convolutional neural networks. *Renewable Energy*, *232*, 121055. https://doi.org/10.1016/j.renene.2024.121055

[28] Zheng, J., Liang, Z.-T., Li, Y., Li, Z., & Wu, Q.-H. (2024). Multi-agent reinforcement learning with privacy preservation for continuous double auction-based P2P energy trading. *IEEE Transactions on Industrial Informatics*, *20*(4), 6582–6590. https://doi.org/10.1109/TII.2023.3348823

[29] Zhao, Z., Guo, J., Luo, X., Xue, J., Lai, C. S., Xu, Z., & Lai, L. L. (2020). Energy transaction for multi-microgrids and internal microgrid based on blockchain. *IEEE Access*, *8*, 144362–144372. https://doi.org/10.1109/ACCESS.2020.3014520

[30] Zhang, D., Yuan, Q., Meng, L., Xia, R., Liu, W., & Qin, C. (2026). Reinforcement learning for single-agent to multi-agent systems: From basic theory to industrial application progress, a survey. *Artificial Intelligence Review*, *59*(2), 46. https://doi.org/10.1007/s10462-025-11439-9

[31] Qiu, D., Chen, T., Strbac, G., & Bu, S. (2023). Coordination for multienergy microgrids using multiagent reinforcement learning. *IEEE Transactions on Industrial Informatics*, *19*(4), 5689–5700. https://doi.org/10.1109/TII.2022.3168319

[32] Wu, Y., Zhao, T., Yan, H., Liu, M., & Liu, N. (2023). Hierarchical hybrid multi-agent deep reinforcement learning for peer-to-peer energy trading among multiple heterogeneous microgrids. *IEEE Transactions on Smart Grid*, *14*(6), 4649–4665. https://doi.org/10.1109/TSG.2023.3250321

[33] Giannuzzo, L., Minuto, F. D., Schiera, D. S., & Lanzini, A. (2024). Reconstructing hourly residential electrical load profiles for renewable energy communities using non-intrusive machine learning techniques. *Energy and AI*, *15*, 100329. https://doi.org/10.1016/j.egyai.2023.100329

[34] Gamage, G., Mills, N., de Silva, D., Manic, M., Moraliyage, H., Jennings, A., & Alahakoon, D. (2024). Multi-agent RAG chatbot architecture for decision support in net-zero emission energy systems. In *2024 IEEE International Conference on Industrial Technology*, 1–6. https://doi.org/10.1109/ICIT58233.2024.10540920

[35] Osipov, E., Kahawala, S., Haputhanthri, D., Kempitiya, T., de Silva, D., Alahakoon, D., & Kleyko, D. (2024). Hyperseed: Unsupervised learning with vector symbolic architectures. *IEEE Transactions on Neural Networks and Learning Systems*, *35*(5), 6583–6597. https://doi.org/10.1109/TNNLS.2022.3211274

[36] Li, Q., Xu, Y., Chew, B. S. H., Ding, H., & Zhao, G. (2022). An integrated missing-data tolerant model for probabilistic PV power generation forecasting. *IEEE Transactions on Power Systems*, *37*(6), 4447–4459. https://doi.org/10.1109/TPWRS.2022.3146982

[37] Lai, Z., Zhang, D., Li, H., Jensen, C. S., Lu, H., & Zhao, Y. (2023). LightCTS: A lightweight framework for correlated time series forecasting. *Proceedings of the ACM on Management of Data*, *1*(2), 125. https://doi.org/10.1145/3589270

[38] Burkart, N., & Huber, M. F. (2021). A survey on the explainability of supervised machine learning. *Journal of Artificial Intelligence Research*, *70*, 245–317. https://doi.org/10.1613/jair.1.12228

[39] Canese, L., Cardarilli, G. C., di Nunzio, L., Fazzolari, R., Giardino, D., Re, M., & Spanò, S. (2021). Multi-agent reinforcement learning: A review of challenges and applications. *Applied Sciences*, *11*(11), 4948. https://doi.org/10.3390/app11114948

[40] Zhou, L., Zheng, Y., Zhao, Q., Xiao, F., & Zhang, Y. (2022). Game-based coordination control of multi-agent systems. *Systems & Control Letters*, *169*, 105376. https://doi.org/10.1016/j.sysconle.2022.105376

[41] Khetarpal, K., Riemer, M., Rish, I., & Precup, D. (2022). Towards continual reinforcement learning: A review and perspectives. *Journal of Artificial Intelligence Research*, *75*, 1401–1476. https://doi.org/10.1613/jair.1.13673

[42] Gulino, C., Fu, J., Luo, W., Tucker, G., Bronstein, E., Lu, Y., ..., & Sapp, B. (2023). Waymax: An accelerated, data-driven simulator for large-scale autonomous driving research. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 7730–7742.

[43] Ning, Z., & Xie, L. (2024). A survey on multi-agent reinforcement learning and its application. *Journal of Automation and Intelligence*, *3*(2), 73–91. https://doi.org/10.1016/j.jai.2024.02.003

[44] Wen, M., Kuba, J. G., Lin, R., Zhang, W., Wen, Y., Wang, J., & Yang, Y. (2022). Multi-agent reinforcement learning is a sequence modeling problem. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, 16509–16521.

[45] Albrecht, S. V., Christianos, F., & Schäfer, L. (2024). *Multi-agent reinforcement learning: Foundations and modern approaches*. USA: MIT Press.

[46] Samadi, E., Badri, A., & Ebrahimpour, R. (2020). Decentralized multi-agent based energy management of microgrid using reinforcement learning. *International Journal of Electrical Power & Energy Systems*, *122*, 106211. https://doi.org/10.1016/j.ijepes.2020.106211

[47] Zhu, C., Ye, D., Zhu, T., & Zhou, W. (2025). The evolution of cooperation in continuous dilemmas via multi-agent reinforcement learning. *Knowledge-Based Systems*, *315*, 113153. https://doi.org/10.1016/j.knosys.2025.113153

[48] Fang, B., Wang, Q., Wang, H., Yu, K., & Wang, Z. (2026). PMARL: Multi-agent reinforcement learning in large-scale systems. *ACM Transactions on Intelligent Systems and Technology*, *17*(3), 73. https://doi.org/10.1145/3799229

[49] Zhong, Y., Kuba, J. G., Feng, X., Hu, S., Ji, J., & Yang, Y. (2024). Heterogeneous-agent reinforcement learning. *Journal of Machine Learning Research*, *25*(32), 1–67.

[50] Ge, Y., Xie, J., Chang, J., & Feng, S. (2025). A multi-objective deep reinforcement learning method for intelligent

scheduling of wind-solar-hydro-battery complementary generation systems. *International Journal of Electrical Power & Energy Systems*, *167*, 110635. https://doi.org/10.1016/j.ijepes.2025.110635

[51] Liu, P., Chokwitthaya, C., Olofsson, T., & Lu, W. (2026). Demand response optimization incorporating thermal comfort in single-family houses with on-site generation: A systematic review. *Applied Energy*, *406*, 127305. https://doi.org/10.1016/j.apenergy.2025.127305

[52] Liu, W., Hu, W., Jing, W., Lei, L., Gao, L., & Liu, Y. (2025). Learning to model diverse driving behaviors in highly interactive autonomous driving scenarios with multi-agent reinforcement learning. *IEEE Systems Journal*, *19*(1), 317–326. https://doi.org/10.1109/JSYST.2025.3528976

How to Cite: De Silva, D., Kempitiya, T., Madhusanka, N., Rathnayaka, P., Mills, N., Jennings, A., & Manic, M. (2026). Multi-agent Reinforcement Learning with Clustering and Forecasting for Optimized Energy Sharing in Microgrids. *Journal of Computational and Cognitive Engineering*, *5*(2), 187–201. https://doi.org/10.47852/bonviewJCCE62027858