Multi-Agent Deep Hedging: Benchmarking Prosumer Strategies on Electricity Trading Platforms

Nicolas Eschenbaum, Nicolas Greber and Oleg Szehr*

July 7, 2025

Abstract

This paper presents Multi-Agent Deep Hedging (MADH), a computational framework that extends deep learning-based financial planning and pricing techniques to markets with multiple strategic participants. Unlike traditional deep hedging, which assumes a passive market environment, MADH models the market as a multi-agent system by augmenting gradient-based optimization with a differentiable market-clearing mechanism, enabling agents to internalize their market impact and optimize policies through gradient backpropagation. In a detailed case study, we apply MADH to model prosumer behavior on electricity trading platforms. We benchmark decentralized MADH using both synthetic and real-world data against a welfare-maximizing central optimizer. Results demonstrate that decentralized agents robustly converge to high-quality policies, engaging in dynamic hedging and strategic capacity withholding while achieving near-optimal welfare. Moreover, we document that MADH performance scales well to a large setting with 32 agents.

*Eschenbaum: Swiss Economics, Ottikerstr. 7, 8006 Zurich, Switzerland (nicolas.eschenbaum@swiss-economics.ch); Greber: Swiss Economics, Ottikerstr. 7, 8006 Zurich, Switzerland (nicolas.greber@swiss-economics.ch); Szehr: Dalle Molle Institute for Artificial Intelligence, Via la Santa 1, 6962 Viganello, Switzerland (oleg.szehr@idsia.ch). We gratefully acknowledge financial support from the Swiss Federal Office of Energy through grant no. SI/502525

1 Introduction

Most hedging and trading methods treat market prices as exogenous, assuming that rebalancing and trading decisions can be optimized without considering their effect on the price dynamics. But in many environments there are only few market actors or a large agent whose quotes moves prices, in which case this assumption breaks down. To optimize hedging decisions in such environments, we introduce Multi-Agent Deep Hedging (MADH), a novel architecture that extends financial deep hedging techniques to multi-agent settings. We apply our architecture to peer-to-peer electricity trading among prosumers, which can be modeled as a stochastic (Markov) game. We benchmark MADH in this environment, show its ability to scale to many agents, and analyse the equilibrium battery-charging and curtailment strategies that emerge.

1.1 Multi-Agent Deep Hedging

Deep Hedging (DH) is a pioneering machine learning approach introduced in Buehler et al. (2019) that has seen widespread success in industry applications, due to the simplicity of its architecture and its scalability to high-dimensional markets involving large numbers of assets. DH models multi-step financial planning problems using a stack of shallow neural networks (NNs), or alternatively, a recurrent NN or a long short-term memory (LSTM) network. Each network in the stack learns the policy corresponding to investment decisions at a specific time step, with gradients propagated through the entire architecture to optimize a utility objective over the investment horizon. This approach has found broad applications beyond traditional finance, including in balance sheet optimization (Krabichler and Teichmann, 2023) and strategic planning in energy consumption and storage (Curin et al., 2021).

We extend the DH framework to a multi-agent setting. Specifically, we equip each agent with a NN policy and model the market-clearing price as a function of all agents' actions. This enables each agent to internalize its individual price impact via joint gradient updates, leveraging standard backpropagation. Rather than treating the market as a passive or fixed environment, the MADH framework endogenizes market dynamics – explicitly modeling how agents' decisions shape prices and how those prices, in turn, influence agent incentives.

1.2 Application

We apply our MADH architecture to prosumer electricity trading platforms (PEPs) to study its performance. With the growing adoption of distributed energy resources, such as photovoltaics and batteries, individual consumers are increasingly becoming producers and need to steer their energy assets. PEPs are seen as one promising approach that enables individuals to trade their energy production and needs by creating digital marketplaces on which prosumers can exchange energy bilaterally instead of relying on the grid.

However, PEPs are complex strategic games for prosumers. Formally, the environment corresponds to a partially observable Markov game (POMG) with incomplete information, where each agent decides on battery charging and photovoltaic curtailment to minimize daily energy costs. The choice of battery charge in particular offers a dynamic hedge against two types of uncertainty. First, it lets the agent buffer future shocks to her own net demand by shifting energy across time. Second, because the platform prices react endogenously to the joint production, demand, and storage choices of all agents, the same intertemporal shift also hedges the systemic price risk created by the stochastic generation and demand of others. Hedging the battery charge optimally over time is therefore a complex task for each prosumer and in practice prosumers cannot be expected to become expert energy traders; they are likely to instead rely on algorithms and automated tools to manage their assets.

Our MADH application allows for a systematic analysis of prosumer incentives and behavior and provides a tool for the assessment of emerging market regulations for energy trading platforms. Regulatory policies can be represented in the POMG as the "rules of the game." Our MADH framework then allows analyzing the emergent agent behavior and its impact on economic welfare and grid stability. In this paper, we evaluate MADH by comparing the decentralized learning of MADH, where each agent independently optimizes its policy while internalizing price effects, against a centralized planner that jointly optimizes all actions. This provides a benchmark to quantify the potential efficiency loss from decentralization and strategic behavior.

1.3 Experimental Findings

Our results demonstrate that decentralized agents that internalize their effect on market prices (i.e., with access to the full computation graph) learn sophisticated, strategically-aware policies. First, agents optimize battery steering dynamically by charging their bat-

tery during the day (when photovoltaic production depresses market prices) and discharging in the evening (when market prices are high). Second, agents learn to strategically withhold capacity by not discharging their battery fully by the end of the day.[1] That is, agents learn to 'coordinate' to restrict market supply in late evening hours in order to avoid market prices dropping to the feed-in tariff (the lower bound on the market price). Third, this emergent coordination breaks down when agents are not fully price-aware, i.e. when they do not have access to the full computation graph. In addition, while agents with access to the full computation graph achieve near-optimal efficiency—total welfare is almost identical to welfare achieved by the central optimizer—, when agents instead do not internalize their price effect, we observe a notable drop in efficiency. Fourth, we compute individual agents' regret and consistently find regret values of less than 1% relative to the learned policy, indicating that agents converge to approximate Nash equilibria.

We further document that when agents' battery endowments are heterogeneous, decentralized optimization results in a redistribution of surplus from agents without a battery to agents with a battery while total welfare is again almost identical to the central optimizer. Because all agents can always cover their entire demand by drawing from the grid at grid prices when needed, capacity withholding results in higher platform prices, but not a decrease in total quantity demanded. As a result, when agents with a battery strategically move market prices, individual surplus is transferred between agents without changing total welfare.

We then show that MADH scales to large settings. Each agent's demand and supply profile is synthesized from anonymized smart-meter data of households and a publicly available, physics-based photovoltaic model. Agents robustly learn and converge to stable, strategically-sophisticated policies across experiments. We also observe that agents with a battery substantially outperform prosumers without a battery, particularly if the battery size is large. For some combinations of profiles with a large battery, agents can in fact achieve negative cost, i.e. a positive profit from participation on the platform. The results show that MADH combines strong scalability with practical simplicity and is potentially a powerful framework to study behavior and optimize decision-making in multi-agent systems.

[1] Note that this is a finite game and thus discharging fully in the final hour is strictly profit-maximizing at any positive market price, ceteris paribus.

Related Literature

This paper is situated at the intersection of multiple strands of literature. As discussed above, we build on the development of Deep Hedging for the hedging of financial portfolios in (Buehler et al., 2019) and related works. We also add to an existing literature that demonstrates the effectiveness of MARL for managing local energy resources under uncertainty, particularly for load scheduling, storage coordination, and price-sensitive behavior in microgrids (e.g., Roesch et al., 2020; Samadi et al., 2020; Gao et al., 2021). However, this literature emphasizes system-level energy costs or improved voltage control and typically does not study strategic behavior of individual agents.[2] Examples include the use of Multi-Agent Deterministic Policy Gradient (MADDPG) to learn bidding strategies (Samende et al., 2022) and Factored Multi-Agent Centralized Policy Gradients (FACMAC), which uses a factored critic to improve scalability while managing network voltage constraints (Charbonnier et al., 2025). Feng and Liu (2025), for instance, propose a consensus-based MARL algorithm where agents share and average their critic network parameters to cooperatively align their policies and ensure network feasibility. Our work employs neither a Centralized Learning and Decentralized Execution approach nor requires direct agent communication.

In addition, there is an extensive literature on the design and taxonomy of prosumer trading markets (e.g., Khorasany et al., 2018; Sousa et al., 2019; Rodrigues et al., 2020; Zhou et al., 2020; Domènech Monfort et al., 2022; Bukar et al., 2023; Hönen et al., 2023a; Tushar et al., 2021; Tsaousoglou et al., 2022; Tushar et al., 2023). Our analytical framework adds to this literature by providing a large-scale simulation tool to investigate how decentralized policies emerge and perform under varying market designs and regulatory conditions. Our findings also contribute to the ongoing discussions on the governance of energy platforms (Weiller and Pollitt, 2013; Rosen and Madlener, 2016; Kloppenburg and Boekelo, 2019; Capper et al., 2022).

The strategic behavior of agents that we document is also connected to the literature on algorithmic collusion by sellers (e.g., Calvano et al., 2020; Eschenbaum et al., 2022), in particular on online marketplaces (e.g., Brero et al., 2022). This literature focuses on price-setting in stylized oligopoly games with standard learners (Q-learning in particular), while we study quantity-setting in a tailored application to energy platforms with a novel

[2]Though MARL has previously been applied to automate agent bidding strategies in P2P trading environments (e.g., Zhang et al., 2018b,a; Qiu et al., 2021).

MADH architecture.

1.4 Overview

The paper is structured as follows. Section 2 develops the Multi-Agent Deep Hedging (MADH) architecture. Section 3 introduces our application of prosumer trading as a POMG, while section 4 outlines the computational framework, including data generation and training procedures. Section 5 then presents our experimental findings. Finally, Section 6 concludes.

2 Multi-Agent Deep Hedging

We propose Multi-Agent Deep Hedging (MADH), a method designed for finite-horizon, partially observable Markov Games (POMGs) (see Section 3.1). MADH is formulated as a multi-agent reinforcement learning (MARL) framework, and we introduce two key variants: (i) a Decentralized Training and Decentralized Execution (DTDE) setting, in which each agent learns an independent policy and acts autonomously based on their local observations, and (ii) a Centralized Training and Centralized Execution (CTCE) setting, where a single global policy is trained using full access to the global state which outputs joint actions for all agents.

Our architecture is motivated by the simplicity and proven success of MARL in industrial applications (e.g., Mnih et al., 2015; Silver et al., 2016). However, standard RL algorithms often struggle in environments characterized by high stochasticity. These challenges are amplified in MARL settings due to increased complexity and non-stationarity (Lowe et al., 2017). While our MADH architecture leverages environmental stochasticity to encourage exploration—akin to techniques used in continuous control RL methods such as DDPG (Lillicrap et al., 2015)—this passive approach to exploration is often insufficient. Therefore, explicit stochastic policies are also incorporated into MADH to enhance exploration and robustness.

2.1 Decentralized Training and Execution

In the DTDE setting, each agent $i \in \mathcal{P}$ maintains its own policy $\pi^{\theta^i} : \mathcal{O}^i \to \mathcal{A}^i$, which maps local observations to actions. These policies are parameterized as neural networks

and trained independently using only local information. Agents do not observe the global state, coordinate with others, or share information—neither during training nor execution.

The policies π^{θ^i} consist of a feedforward neural network with L hidden layers and ReLU activations, followed by a tanh:[3]

$$h^0 = o^i \in \mathcal{O}^i, \quad h^\ell = \sigma(W_\ell h^{\ell-1} + b_\ell), \quad \pi^{\theta^i}(x) = \tanh(W_L h^{L-1} + b_L), \quad \ell = 1, \ldots, L-1 \tag{1}$$

where $\{W_\ell, b_\ell\}$ are agent-specific trainable parameters. Recurrent alternatives such as LSTMs may also be used in partially observable settings.

Each agent acts independently based on its private observations. These actions are submitted to a central market mechanism, which computes prices and updates the environment. The resulting global state induces new observations for the next individual agent decisions. Figure 1.

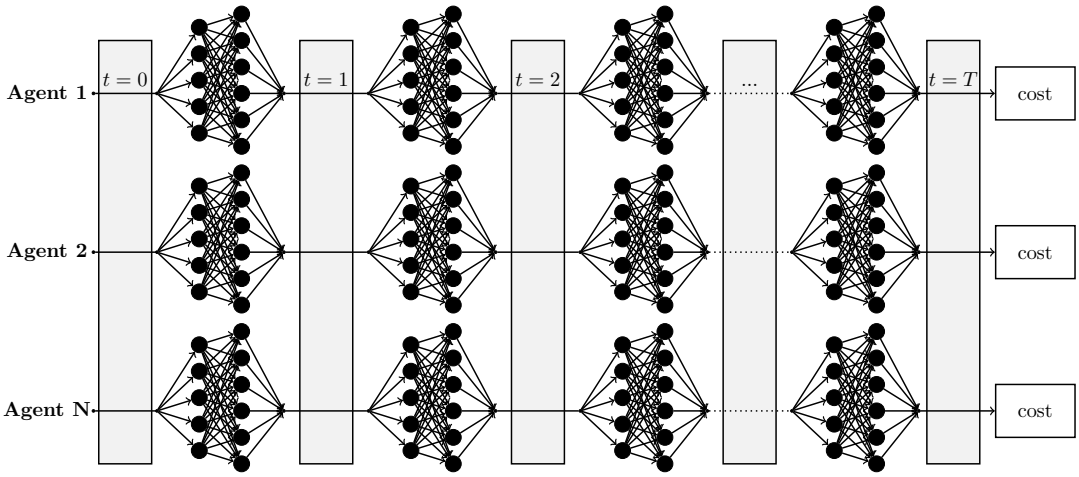


Figure 1: Each agent has an individual neural network policy that sequentially maps its private observations to actions. The transition functions (shown in grey) determine the observations available to each agent.

Policies are trained batch-wise via backpropagation through the full unrolled trajectory of the environment. We distinguish two cases in DTDE: (i) backpropagation with full computation graph, where agents engage in gradient descent through the environemnt

[3]The tanh function is used to bind the network's output to a specific range. In our application, this continuous output corresponds to two actions: the charging/discharging rate and the percentage of PV production to curtail.

updates, and (ii) backpropagation without full computation graph, where agents only consider their action and not its influence on the environment (in our case the market price). Each trajectory records observations, actions, and incurred costs, or

$$\tau_b^i = \{(o_{t,b}^i, a_{t,b}^i, c_{t,b}^i)\}_{t=0}^T, \tag{2}$$

where $c_{t,b}^i$ is the cost incurred by agent i at time t in batch b. After collecting a batch, the average cost is computed as

$$\bar{C}^i \;=\; \frac{1}{B}\sum_{b=1}^{B}\sum_{t=0}^{T} c_{t,b}^i, \tag{3}$$

and agent parameters are updated via backpropagation, or

$$\theta^i \;\leftarrow\; \theta^i \;-\; \alpha\,\nabla_{\theta^i}\bar{C}^i. \tag{4}$$

Exploration is encouraged via randomization in actions. We use a decaying ϵ-greedy scheme in which agents sample a random action uniformly from their action space with probability ϵ and otherwise follow their policy output. This ensures broad exploration early in training and shifts toward exploitation over time. Algorithm 1 summarizes training for the DTDE setting.

Algorithm 1 MADH with Decentralized Learning & Execution

Input: Agents $\{1, \ldots, N\}$, horizon T, batch size B, episodes K, learning rate α
for each episode $k = 1$ to K **do**
  Update probability of random action p_ε
  Sample B days in parallel:
  for training horizons $b = 1$ to B **do**
    Initialize state s_0, set cumulative costs $C_b^i \leftarrow 0$
    for $t = 0$ to T **do**
      Each agent i computes $a_t^i = \pi^{\theta^i}(o_t^i)$
      With probability ρ_k replace each component of a_t^i with random draws $\varepsilon \sim U[A]$
      Market clears: $(s_{t+1}, \{o_{t+1}^i\}) f(s_t, (a_t^1, \ldots, a_t^N), \xi_{t+1})$
      Accumulate cost: $C_b^i \mathrel{+}= c_t^i$
    end for
  end for
  $\theta^i \leftarrow \theta^i - \alpha \nabla_{\theta^i} C_b^i, \ \forall i$
end for

2.2 Centralized Learning and Execution

As an efficiency benchmark, we implement a central optimizer with full observability of all agents' states. A single policy $\pi^\theta : \mathcal{S} \to \mathcal{A}_1 \times \cdots \times \mathcal{A}_N$ is trained to minimize the total cost, $\sum_{i=1}^N C^i$, using batch backpropagation. The architecture of the planner mirrors that of the decentralized agents but unlike the individual agents it receives the complete state at each time period. The central optimizer then outputs actions for all agents.

Again, a decaying ϵ-greedy exploration strategy is used during training to mitigate the risk of converging to poor local minima and to promote robustness in continuous action spaces. Importantly, this central optimizer does not maximize individual agent objectives, but rather the aggregate total economic welfare of all agents.[4]

[4]Note that this central optimizer differs from the standard concept of a central planner in economic models. In typical formulations, a central planner selects the allocation of resources to all agents in order to maximize social welfare. In our case, the central optimizer instead selects the players' actions in order to maximize total welfare.

Algorithm 2 MADH with Centralized Learning & Execution

Input: Number of agents N, horizon T, batch size B, episodes K, learning rate α
for each episode $k = 1$ to K **do**
  Update probability of random action p_ε
  Sample B days in parallel:
  for training horizons $b = 1$ to B **do**
    Initialize global state s_0, set cumulative cost $C_b \leftarrow 0$
    for $t = 0$ to T **do**
      Compute joint action $a_t = \pi^\theta(s_t)$
      With probability ρ_k, replace each component of a_t with a random variable $\varepsilon \sim U[A]$
      Market clears: $s_{t+1} = f(s_t, a_t, \xi_{t+1})$
      Accumulate cost: $C_b \mathrel{+}= \sum_{i=1}^{N} c_t^i$
    end for
  end for
  Update policy: $\theta \leftarrow \theta - \alpha \nabla_\theta C_b$
end for

Figure 2 shows a centralized optimizer in which a single neural network receives the complete system state per period and outputs actions for all agents.

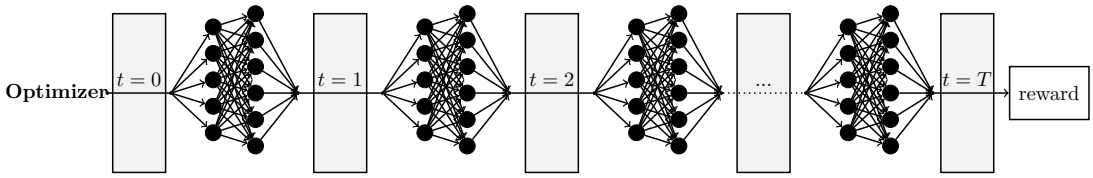


Figure 2: Centralized optimizer with joint action planning.

3 Model Setup

We apply our MADH architecture to peer-to-peer (P2P) trading on prosumer electricity trading platforms. We model the market-clearing, price formation, and policy-learning mechanisms as a partially observable Markov game (Littman, 1994; Eschenbaum et al., 2022) in which prosumer agents interact over a finite horizon. Throughout, all energy-

related variables are measured in kilowatt-hours (kWh) per trading period. Because we discretize time into hourly intervals, 1 kWh is both a flow (one hour of power) and the unit of stock for battery capacity.

3.1 Markov Game

Formally, the game is given by

$$\mathcal{G} = \left(\mathcal{P}, \mathcal{T}, \{\mathcal{A}^i\}, \{\mathcal{O}^i\}, \mathcal{S}, \mathcal{T}^{\text{state}}, \{r^i\}\right),$$

with agents $i \in \mathcal{P}$, periods $t \in \mathcal{T} = \{1, \ldots, T\}$, joint state $s_t \in \mathcal{S}$, and stochastic transition kernel $\mathcal{T}^{\text{state}}(s_{t+1} \mid s_t, a_t)$. Because each agent observes only a private signal $o_t^i \in \mathcal{O}^i$, the game features imperfect information. Policies $\pi^i : (o_{1:t}^i, a_{1:t-1}^i) \mapsto a_t^i$ must therefore cope with non-stationarity induced by the learning behavior of others. Our framework models a general-sum game or mixed game (e.g. see (Zhang et al., 2021) without imposing restrictions on the agents' objectives (Hu and Wellman, 2003; Littman et al., 2001).

3.2 Agents

We assume that there are N prosumers, $\mathcal{P} = \{1, \ldots, N\}$, each endowed with photovoltaics (PV) and a battery of capacity $\overline{B^i} \in \mathbb{R}_+$. At the beginning of period t, agent i learns her stochastic PV generation $g_t^i \in [0, \overline{G^i}]$, stochastic inelastic demand $d_t^i \in [0, \overline{D^i}]$, and current battery state $B_t^i \in [0, \overline{B^i}]$. She chooses a battery charge (positive) or discharge (negative) $b_t^i \in [-B_t^i, \overline{B^i} - B_t^i]$ and a curtailment share $\kappa_t^i \in [0, 1]$. Battery dynamics follow

$$B_{t+1}^i = B_t^i + b_t^i(\mathbf{1}_{\{b_t^i \geq 0\}}\eta^{\text{ch}} - \mathbf{1}_{\{b_t^i < 0\}}(\eta^{\text{dis}})^{-1}), \quad \eta^{\text{ch}}, \eta^{\text{dis}} \in (0, 1], \tag{5}$$

and net injection to the platform is $x_t^i = d_t^i + b_t^i - (1 - \kappa_t^i)g_t^i$, positive for imports and negative for exports (see e.g., Lara et al., 2018; Guerrero et al., 2020; Hönen et al., 2023b). Each agent observes (B_t^i, d_t^i, g_t^i, t) but neither others' battery states nor their chosen actions, making $\mathcal{G}$ a partially observable Markov game (POMG).

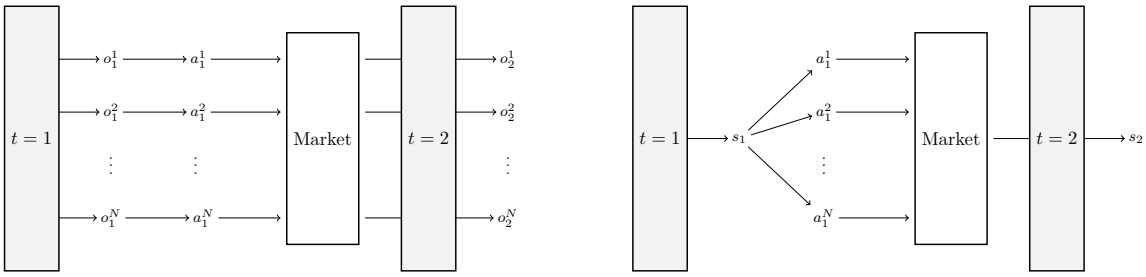
3.3 Environment

The platform aggregates injections, $S_t = \sum_{i=1}^N \max\{-x_t^i, 0\}$ and $D_t = \sum_{i=1}^N \max\{x_t^i, 0\}$, and posts volume-weighted sell and buy prices p_t^S and p_t^D (denoted in €/kWh) that satisfy

the corridor $p^E \leq p_t^S \leq p_t^D \leq p^I$, where p^E and p^I are exogenous feed-in and retail tariffs.[5] Any admissible clearing rule is a mapping $f : (\mathbf{x}_t, p^E, p^I) \mapsto (p_t^S, p_t^D)$ with residual imbalances settled against the grid. Agent i's one-period cost c_t^i and total cost C^i are

$$c_t^i = \begin{cases} x_t^i \, p_t^D, & x_t^i > 0, \\ x_t^i \, p_t^S, & x_t^i < 0, \end{cases} \qquad C^i = \sum_{t=1}^{T} c_t^i, \tag{6}$$

respectively. Figure 3 shows the interaction of the two algorithms DTDE and CTCE with the market model over one period of the game.



(a) Decentralized Algorithm (DTDE). (b) Centralized Algorithm (CTCE).

Figure 3: Interaction of the decentralized and centralized algorithms with the market model.

3.4 Pricing Mechanism

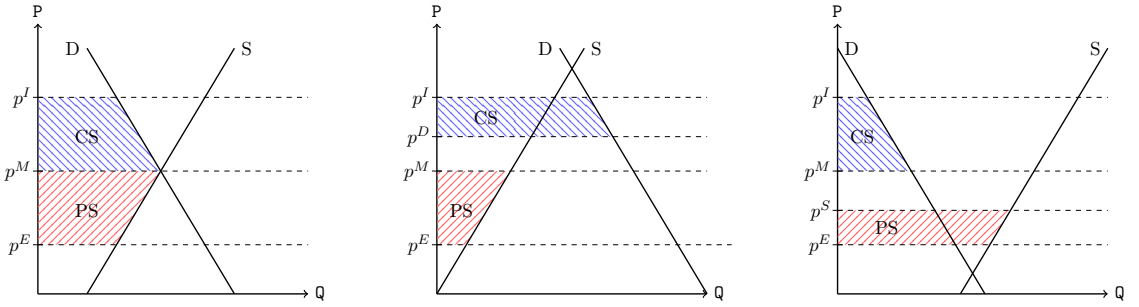
For our experiments, we use the mid-market-rate (MMR) rule as described by Long et al. (2017). The benchmark price is $p^M = \frac{1}{2}(p^I + p^E)$. If $S_t = D_t$ at p^M the market clears at $p_t^S = p_t^D = p^M$; otherwise the short side trades at p^M and the long side's price adjusts linearly toward the relevant grid tariff:

$$p_t^S = \begin{cases} p^M, & S_t \leq D_t, \\ \dfrac{p^M D_t + p^E (S_t - D_t)}{S_t}, & S_t > D_t, \end{cases} \qquad p_t^D = \begin{cases} \dfrac{p^M S_t + p^I (D_t - S_t)}{D_t}, & D_t > S_t, \\ p^M, & D_t \leq S_t. \end{cases} \tag{7}$$

MMR is continuous in (p^I, p^E) and differentiable in `PyTorch`, enabling gradient-based policy search. By avoiding the price collapse to p^I or p^E that characterises other popular discrete

[5] Note that participation constraints for agents imply the price corridor, as across jurisdictions prosumers are always allowed to draw from the grid at the retail price and sell to the grid at the feed-in tariff.

mechanisms in the literature, MMR allows for smoother price movements and changes to surplus allocation, limiting sudden jumps in incentives for agents.



(a) $D(p^M) = S(p^M)$: uniform price at p^M.

(b) $D(p^M) > S(p^M)$: price gap favors suppliers.

(c) $D(p^M) < S(p^M)$: price gap favors buyers.

Figure 4: Surplus distribution under the Mid-Market Rate (MMR) pricing rule, driven by supply-demand imbalances.

Figure 4 shows how MMR (partially) avoids corner solutions by allocating rent between consumer surplus (CS) and producer surplus (PS). When demand exceeds supply (Figure 4b), sellers receive the highest feasible price ($p^S = p^M$), while buyers pay a premium ($p^D > p^M$). In the inverse case (Figure 4c), buyers pay p^M, and sellers receive less. As a result, there are smoother transitions and partial surplus sharing even in imbalanced markets, unlike discrete-clearing mechanisms that collapse to p^I or p^E.

4 Experimental Setup

This section describes the data sources, learning architecture, simulation scenarios and evaluation metrics used to assess the Multi-Agent Deep Hedging (MADH) framework.

4.1 Data sets

We make use of two data sources. One, we develop a synthetic data set that provides full experimental control and simplicity. Each day is divided into $T = 24$ hourly steps. Every agent receives three exogenous inputs per time horizon: demand, PV generation, and initial state of charge. Clear-sky PV is a half-sine curve from 6am to 8pm and and demand follows a bimodal profile with peaks at 8am and 7pm. Cloud cover is an AR(1) factor truncated to $[0.4, 1.0]$. Daily demand is rescaled by Beta$(4, 4)$ (± 30 %) and perturbed by $\mathcal{N}(0, 0.15^2)$.

Two, we build an empirical data set that combines photovoltaic (PV) output with anonymised household load traces from smart-meter data. First, the PV output is generated with `pvlib-python 0.11`, an open-source Python library for modelling photovoltaic energy systems (Anderson et al., 2023).[6] The underlying weather data is taken from the Photovoltaic Geographical Information System (PGIS) and a typical meteorological year for Zurich (Switzerland).[7]. The PGIS database returns 8 760 hourly records which are then timezone-converted to `Europe/Zurich`. Second, the household demand profiles are taken from the anonymised CKW smart-meter archive (Centralschweizerische Kraftwerke AG (CKW), 2025). We only use meters with 10–20 kWh average daily use and ≥ 30 clean days are retained, giving hourly 24-element vectors indexed by date. We augment this raw data by adding noise for training. Specifically, for each simulation day the loader adds i.i.d. noise $\mathcal{U}(0.7, 1.3)$.

Note that unless otherwise indicated, p^I is set at 0.3 €/kWh and p^E is 0.05 €/kWh, battery charge and discharge efficiencies are $\eta^{ch} = \eta^{dis} = 0.95$, and initial battery states are sampled from $U[0, \overline{B^i}]$.

4.2 MADH Model Architecture and Training

Each agent's policy is a two-layer feed-forward network with 32 hidden units and ReLU activations; the output layer uses tanh to bound actions. The central optimizer benchmark employs a larger network (128 units per layer in setups (i) and (ii), 512 in setup (iii), see Section 5). Networks are trained with Adam stochastic gradient descent ($\alpha = 10^{-3}$, batch 64) over $K = 120$ episodes. Exploration follows an ε-greedy schedule starting at 0.5 and decaying by factor 0.95 per episode. All experiments run under Python 3.9.7 and PyTorch 2.2.1 on Apple M1 (CPU).

4.3 Scenarios

We evaluate MADH in three different scenarios.

[6] We employ ASHRAE angle-of-incidence (AOI) optics, which models the angular-dependent transmission losses using a single-parameter function, SAPM (Sandia Array Performance Model) temperature modeling, which determines cell temperature from effective irradiance and ambient conditions using empirically derived coefficients, and a 96% inverter efficiency. The PV system parameters are randomized within realistic ranges: tilt $[25°, 35°]$ and azimuth $[165°, 195°]$ (south-facing) $\pm 15°$.

[7] Specifically, we obtain the data for the coordinates $47.38°\text{N}, 8.54°\text{E}$ via the `get_pvgis_tmy` Python library (Jensen et al., 2023)

(i) **Two Homogeneous Agents:** Two identical agents, each endowed with a battery of rated capacity ($\overline{B} = 8$ kWh). The load and photovoltaic (PV) profiles are sourced from the synthetic data set. This setup serves as a baseline to understand agent incentives and test for strategic behavior.

(ii) **Three Heterogeneous Agents:** Three agents who are identical in all respects except for their battery storage capacity. The load and PV profiles are as before sourced from the synthetic data set. One agent has no storage (0 kWh), the second has a medium-sized battery (4 kWh), and the third has a large battery (8 kWh). This scenario is designed specifically to study how strategic behavior and economic welfare are affected by asymmetries between participants.

(iii) **32 Heterogeneous Agents:** Large-scale energy community consisting of 32 heterogeneous prosumers. The load and PV profiles of each household are sourced from the real-world data set. Agents are divided into four types based on their asset configurations: (i) agents with 8 kWh battery storage and PV generation, (ii) agents with 4 kWh batteries and PV, and (iii) agents with generation capacity but without storage and (iv) agents without generation and storage. This scenario tests the stability and scalability of our MADH architecture and robustness of previous findings.

4.4 Outcome Measures

To evaluate efficiency of the agents' learned policies with MADH, we consider a normalization of the agents' surplus relative to the surplus under the central optimizer denoted by Δ and Δ^i respectively, or

$$\Delta = \frac{C_{\text{OS}} - C_{\text{Dec}}}{C_{\text{OS}} - C_{\text{Cent}}} \qquad \Delta^i = \frac{C^i_{\text{OS}} - C^i_{\text{Dec}}}{C^i_{\text{OS}} - C^i_{\text{Cent}}} \tag{8}$$

where C_{Cen} is the total system cost under the centralized planner, C_{Dec} under decentralized coordination (MADH), and C_{OS} under a one-shot Nash strategy.

Note that $\Delta^i > 1$ is possible, as an individual agent may achieve lower cost under decentralized coordination than in the centralized solution, due to the central planner optimizing for system-wide welfare rather than individual profit. However, by construction, $\Delta \leq 1$, as decentralized coordination cannot outperform the central planner in terms of aggregate welfare.

In addition, to verify that the converged decentralized policies form a stable outcome, we test for an ε-Nash Equilibrium, where ε is a real non-negative parameter. The set of policies of all agents are an equilibrium if no agent can gain more than ε by unilaterally changing its policy while all other agents' policies remain fixed. To test for an ε-Nash Equilibrium, we determine the unilateral deviation regret of agents. For each agent i, we first train the full N-agent system to find the set of MADH equilibrium policies (π^*_{Dec}). Then, we freeze the policies of all agents except i and retrain agent i to find its optimal best-response policy, π^i_{BR}. The regret for agent i, denoted $\mathcal{R}^i$, is the cost improvement it gains by switching to this best-response policy, or

$$\mathcal{R}^i = \frac{C^i_{\text{Dec}} - C^i_{\text{BR}}}{|C^i_{\text{Dec}}|} \tag{9}$$

where C^i_{Dec} is the agent's cost under the original decentralized equilibrium policy, and C^i_{BR} is its cost under the newly computed best-response policy, evaluated against the fixed policies of its opponents.

A regret value $\mathcal{R}^i$ close to zero (i.e., less than ε) implies that the agent's original policy, $\pi^{*,i}_{\text{Dec}}$, was already its (approximate) best response to the policies of the other agents, and thus there is no incentive to deviate. The sum of all individual regrets, ($\sum_i \mathcal{R}^i$), then measures the overall stability of the set of converged policies. A low sum of individual regrets is strong evidence that the MADH algorithm has converged to an ε-Nash Equilibrium.

5 Experimental Findings

This section presents the empirical results from three different scenarios that test the performance, stability, and scalability of MADH. We begin with the simplest possible setup with two homogeneous agents to explore the learning dynamics and policies. We then introduce a third agent and vary the size of the battery among the three players – where one agent has no battery, i.e. a size of zero – to analyze the distributional consequences of strategic behavior. Finally, we study a large setting with 32 agents and demonstrate MADH's scalability as well as the robustness of the previous findings.

5.1 Two Homogeneous Agents

Figure 5 shows the mean episode cost over 120 training episodes for two homogeneous agents with photovoltaic generation and a battery. Both the decentralized learners (in blue) and the central optimizer (red) show rapid cost reduction, with the majority of the learning occurring within the initial 30 episodes. This fast convergence is particularly notable given the non-convex nature of the payoff landscape, because the strategic interaction between agents can introduce plateaus to the loss function that could lead the learner to converge to local minima.

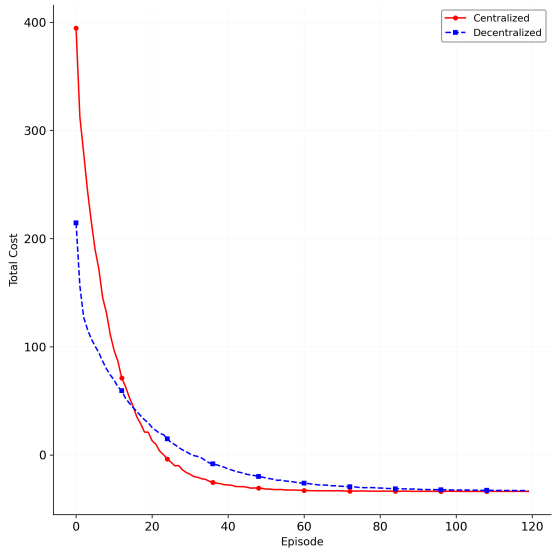


Figure 5: Mean per-episode total costs in the symmetric two-agent market.

Table 1 shows the efficiency of MADH in comparison to the central optimizer and to the (repeated) myopic one-shot Nash equilibrium. We provide results both when the decentralized learners have access to the full computation graph (left) and when they do not (right).[8] We find that when agents have access to the full computation graph, the total decentralized costs are almost indistinguishable from those of the central planner. Specifically, we find that $\Delta = 0.996$ and the total cost is identical for Prosumer 2 and only slightly higher under the decentralized case for Prosumer 1. We also observe a small difference in net grid exports. The central optimizer imports 15.08 kWh compared to 13.78

[8]The former correspond to the convergence results for the decentralized case in Figure 5.

kWh with MADH, and similarly exports increase from 18.07 kWh to 19.57 kWh.

	With Graph			Without Graph		
	Prosumer 1	Prosumer 2	Total	Prosumer 1	Prosumer 2	Total
C_{Cent}	0.782	0.782	1.564	0.782	0.782	1.564
C_{Dec}	0.793	0.782	1.575	0.892	0.893	1.786
C_{OS}	2.254	2.254	4.509	2.254	2.254	4.509
Δ	0.993	1.000	0.996	0.925	0.925	0.925

Table 1: Efficiency and cost in the symmetric two-agent market.

However, when agents do not have access to the full computation graph, they perform less well and we obtain $\Delta = 0.925$ with the total cost rising to 1.79 compared to 1.58.[9] The difference between the performance with the full computation graph compared to without it, stems from the interaction of their total net demand with the clearing mechanism. When the agents have access to the computational graph, they account for the endogenous reaction of the market price to their actions. In the two-agent, symmetric case, this is particularly important in the evening because the net demand in the evening hours for both agents is near-zero.

This implies that any significant deviation by one agent towards increased supply in evening hours would lead to the price dropping to the lower bound p^E, making the deviation unprofitable. This creates a strong implicit coordination mechanism that leads to the agents under MADH achieving virtually the same outcome as the centralized optimizer. As a result, under MADH the agents avoid fully discharging their batteries during the evening hours (in line with the behavior of the centralized optimizer). In effect, the agents strategically withhold capacity from the market. In the case where agents do not have access to the full computational graph in turn, they do not fully internalize their impact on the price and supply more energy in the evening hours, explaining their worse performance.

We further find a regret for agent 0 of 0.49% and for agent 1 of 0.16% in the setting with access to the full computation graph. The sum of regrets is -0.33%. This strongly suggests that the learned strategies are stable and represent a ε-Nash equilibrium. Figure 6 shows the policies of Prosumer 1. We document the learned policy both in the decentralized

[9] In addition, in either case, the agents perform much better compared to the myopic one-shot outcome. Cost per prosumer are around 35% of the cost under one-shot behavior.

MADH case and for the central optimizer. In both settings, higher PV supply leads to stronger charging behavior. There is also a tendency to reduce charging or even discharge as the battery state increases.

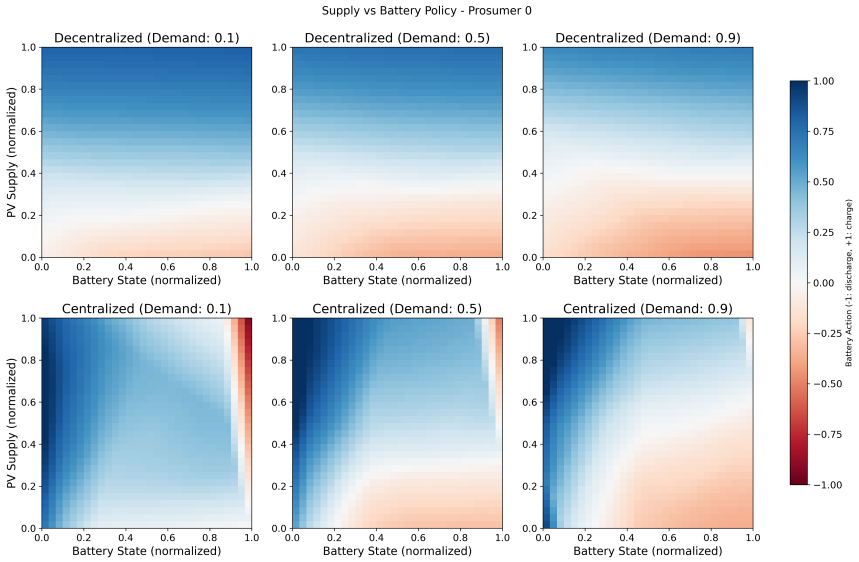
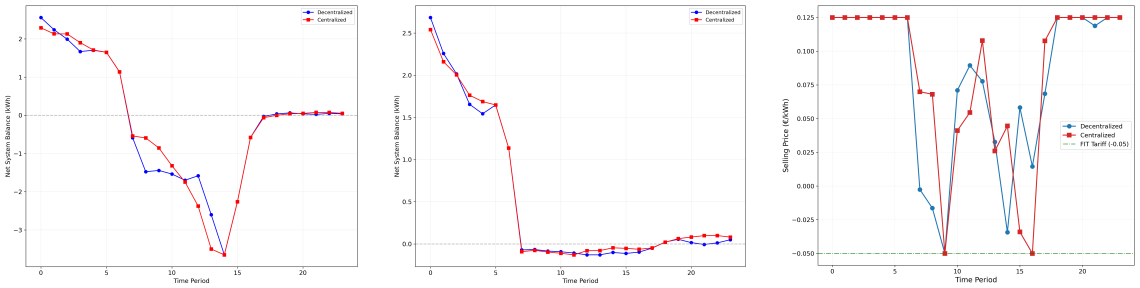


Figure 6: Battery charging policies of Prosumer 1 in the symmetric case. The y-axis shows the PV supply g_t, and the x-axis indicates the given battery state B_{t-1}. The color scale represents the chosen action (charge to discharge on a $[-1, 1]$ interval). Policies are shown for three different demand levels d_t.

We run one additional test in which we set the feed-in tariff to a negative value ($p^E = -0.05$ €/kWh). The results, shown in Figure 7, demonstrate that agents with the full computation graph learn sophisticated responses to avoid penalties. Under a standard positive feed-in tariff, both the centralized and decentralized optimizers allow for negative net balance of the platform (panel (a)). However, when the tariff becomes negative, agents immediately learn to suppress exports by curtailing PV production during periods of high generation. This response is so effective that the system-wide balance is barely negative during high generation, as agents prefer to curtail rather than risk being penalized for exporting surplus energy (panel (b)). Thus, agents do not just react to prices but actively manage their assets to shape market outcomes. As panel (c) shows, with a negative feed-in tariff, the market-clearing selling price in fact rarely falls to the -0.05 €/kWh floor.



(a) Net balance (positive p^E). (b) Net balance (negative p^E). (c) Selling prices (negative p^E).

Figure 7: Impact of a negative feed-in tariff (FIT) on system balance and prices. A negative FIT (b) reverses the near-zero system balance seen with a positive FIT (a) as agents curtail exports. Market-clearing prices (c) remain mostly above the penalty benchmark.

5.2 Three Heterogeneous Agents

Next, we study a setting with three prosumers that are identical in all respects except for their battery sizes: Prosumer 1 has no storage, Prosumer 2 has a medium-sized battery (4kWh), and Prosumer 3 has a large battery (8kWh). This setup allows us to study how asymmetry between agents affects outcomes under MADH and centralized learning, and how these are influenced by the presence of a full computation graph. The main results are documented in Table 2 and Table 3.

	Large (8 kWh)	Medium (4 kWh)	None (0 kWh)	Total
C_{Cent}	1.426	2.030	2.802	6.258
C_{Dec}	1.254	2.006	3.000	6.260
C_{ST}	3.086	3.086	3.086	9.258
Δ	1.104	1.023	0.301	0.999

Table 2: Costs (C) and efficiency (Δ) in the asymmetric three agent market with access to the full computation graph.

We find that prosumers with batteries benefit from decentralized decision-making, while the agent without a battery is better-off under centralized optimization. In fact, under DTDE MADH we find $\Delta = 0.999$, implying that rent is redistributed from the agent without a battery to the agents with the batteries compared to the central optimizer solution. This is again explained by the capacity withholding of individual agents with

batteries in the evening hours. This raises the buying price above the mid-market rate ($p^D > p^M$), making the agent without a battery worse off. However, because the prosumers electricity demanded remains the same irrespective of the price, this increase in prices does not translate to a decrease in quantity sold and is thus a direct transfer of rent from the prosumer without a battery to the prosumers with a battery while total welfare is unaffected.

When agents do not have access to the full computation graphs (Table 3), the effects are more pronounced. When the agents take prices as exogenous, there is a lower efficiency (Δ drops from 0.999 to 0.962).

	Large (8 kWh)	Medium (4 kWh)	None (0 kWh)	Total
C_{Cent}	1.426	2.030	2.802	6.258
C_{Dec}	1.237	2.077	3.059	6.373
C_{ST}	3.086	3.086	3.086	9.258
Δ	1.114	0.956	0.094	0.962

Table 3: Costs (C) and efficiency (Δ) in the asymmetric three agent market without access to the full computation graph.

We also find that the individual regret for each agent is once again very small. The agent with a medium-sized battery was able to slightly reduce its cost (by 0.03%) while the prosumer with the large battery actually performed worse (-0.39%). Total regret is -0.36%. These very small regret values suggest as before that the learned policies with the full computation graph in this setting are near-optimal responses given others' strategies, i.e. agents' strategies form an ε-Nash equilibrium.

5.3 32 Heterogeneous Agents

In order to validate the scalability and practical applicability of our MADH framework, we now study its performance with a large set of agents. We consider 32 heterogeneous agents whose load and generation profiles are derived from real-world data. Our results demonstrate that the decentralized MADH approach is practical at scale. Figures 8a and 8b shows the learning curves (mean and standard deviation) for two types of agents: equipped with a 8kWh battery and photovoltaics (panel (a)) and with a 4 kWh battery and photovoltaics (panel (b)), respectively. We observe robust convergence of the average

cost per episode which steadily decreases before reaching a stable plateau.

The converged policies correspond to economically rational behavior. As illustrated in Figures 8c and 8d, the learned policies implement an arbitrage strategy. Agents systematically accumulate energy in their batteries during the midday period (approximately 10:00–16:00), when high aggregate PV generation depresses the local market price. They then discharge this stored energy in the evening (approximately 18:00–22:00) when demand is high. However, they do not fully discharge their batteries by the end of the day and choose to keep 10-15% of the maximum battery charge. This highlights again the capacity withholding that agents engage in, in order to keep the platform price above the feed-in tariff in evening hours. This shows that agents in our large-scale experiment have consistently learned to account for endogenous price movements in response to their actions, because at the given (positive) market price, ceteris paribus, discharging more is always profitable.

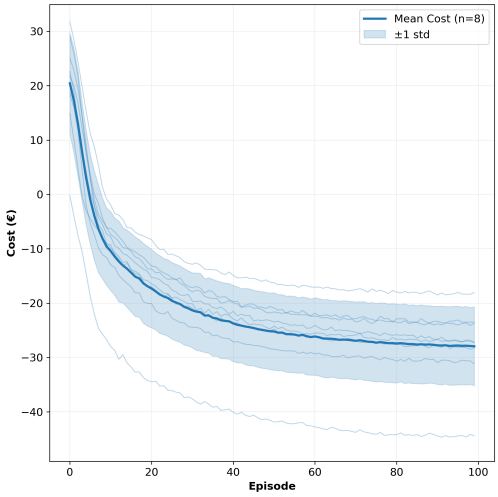
The heterogeneity in agents' battery sizes and photovoltaics leads to systematic differentiation in their economic outcomes. Figure 9 shows the distribution of realized daily costs by agent type. Prosumers with larger batteries and PV systems (i.e., greater flexibility) have operational cost. For types with a battery, the operational cost are even negative on average.

We also observe that the decentralized learning approach yields superior pricing outcomes relative to the myopic one-shot strategy. Specifically, prosumers achieved both higher selling prices (€0.1731 vs. €0.1705) and lower buying prices (€0.2454 vs. €0.2557). Across all deviation tests, no agent was able to reduce their cost by more than 1%. The maximum individual regret observed was 0.55%, indicating that the strategic behavior learned by the agents is indeed stable.

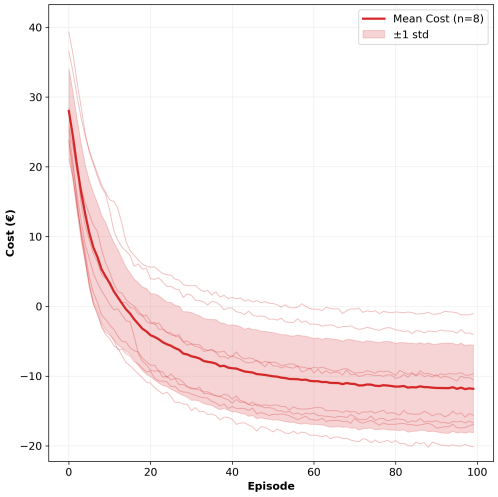
6 Conclusion

This paper introduces Multi-Agent Deep Hedging (MADH), a novel and tractable framework to model and optimize dynamic hedging and trading decisions when market prices are affected by a traders' actions. By extending financial deep hedging methods to a multi-agent setting, MADH provides a powerful methodology to study behavior in complex multi-agent systems. We apply MADH to model and benchmark prosumer strategies on decentralized electricity trading platforms. The simulation experiments show that agents consistently converge to stable and high-quality policies and that MADH robustly scales

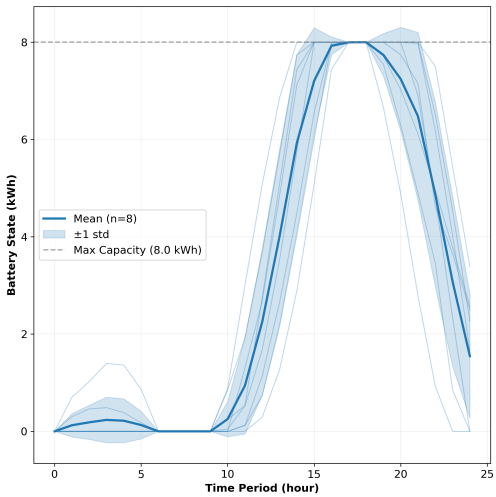
(a) Learning curve: 8 kWh battery + PV



(b) Learning curve: 4 kWh battery + PV



(c) Battery states: 8 kWh battery + PV



(d) Battery states: 4 kWh battery + PV

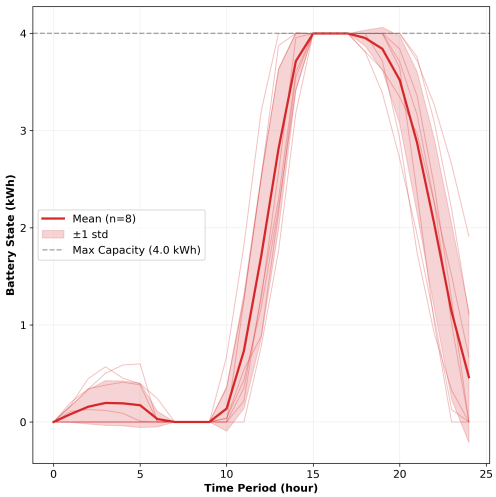


Figure 8: Learning convergence and resulting battery arbitrage behaviour for two prosumer types. **Top row**: average per-episode costs stabilise for agents with larger (left) and smaller (right) batteries. **Bottom row**: corresponding state-of-charge trajectories reveal midday charging and evening discharging once the policy converges.

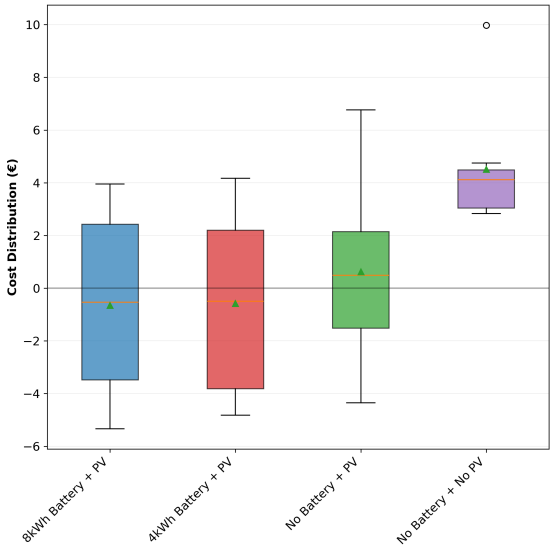


Figure 9: Distribution of daily costs by prosumer type.

to settings with many agents.

We document three main empirical findings. First, when agents have access to the full computation graph, decentralized MADH attains efficiency near-equal to a central optimizer. Total welfare and individual regrets generally differs from the central optimiser by less than 1%, indicating convergence to an -Nash equilibrium. Second, strategic capacity withholding emerges endogenously and redistributes welfare from agents without a battery to agents with a battery. Removing access to the full computation graph, however, breaks this implicit coordination, amplifies price volatility, and reduces system efficiency. Third, MADH scales well. In a 32-agent environment calibrated to smart-meter data, all agent types converge rapidly, maintain sub-percent regret, and exhibit the same economically rational charge–discharge pattern observed in smaller experiments. Our findings demonstrate that MADH provides a tractable, scalable means of analysing strategic behaviour in dynamic markets.

An interesting avenue for future research is to exploit the flexibility of the MADH framework to study different market mechanisms and regulatory choices for prosumer trading platforms or apply it to a different context. Moreover, we have refrained from modeling the underlying electricity grid in more detail in our benchmarking application. Introducing a more realistic grid model would add important trading restrictions and transaction cost

to the learning problem.

References

Anderson, K., Hansen, C., Holmgren, W., Jensen, A., Mikofski, M., and Driesse, A. (2023). pvlib python: 2023 project update. *Journal of Open Source Software*, 8(92):5994.

Brero, G., Mibuari, E., Lepore, N., and Parkes, D. C. (2022). Learning to mitigate ai collusion on economic platforms. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems*, volume 35, pages 37892–37904. Curran Associates, Inc.

Buehler, H., Gonon, L., Teichmann, J., and Wood, B. (2019). Deep hedging. *Quantitative Finance*, 19(8):1271–1291.

Bukar, A. L., Hamza, M. F., Ayup, S., Abobaker, A. K., Modu, B., Mohseni, S., Brent, A. C., Ogbonnaya, C., Mustapha, K., and Idakwo, H. O. (2023). Peer-to-peer electricity trading: A systematic review on currents development and perspectives. *Renewable Energy Focus*.

Calvano, E., Calzolari, G., Denicolò, V., and Pastorello, S. (2020). Artificial intelligence, algorithmic pricing, and collusion. *American Economic Review*, 110(10):3267–97.

Capper, T., Gorbatcheva, A., Mustafa, M. A., Bahloul, M., Schwidtal, J. M., Chitchyan, R., Andoni, M., Robu, V., Montakhabi, M., Scott, I. J., et al. (2022). Peer-to-peer, community self-consumption, and transactive energy: A systematic literature review of local energy market models. *Renewable and Sustainable Energy Reviews*, 162:112403.

Centralschweizerische Kraftwerke AG (CKW) (2025). CKW open data – smart-meter-daten (datensatz a). `https://open.data.axpo.com/%24web/index.html`. Accessed 2025-07-04.

Charbonnier, F., Peng, B., Vienne, J., Stai, E., Morstyn, T., and McCulloch, M. (2025). Centralised rehearsal of decentralised cooperation: Multi-agent reinforcement learning for the scalable coordination of residential energy flexibility. *Applied Energy*, 377:124406.

Curin, N., Kettler, M., Kleisinger-Yu, X., Komaric, V., Krabichler, T., Teichmann, J., and Wutte, H. (2021). A deep learning model for gas storage optimization. *Decisions in Economics and Finance*, 44(2):1021–1037.

Domènech Monfort, M., De Jesús, C., Wanapinit, N., and Hartmann, N. (2022). A review of peer-to-peer energy trading with standard terminology proposal and a techno-economic characterisation matrix. *Energies*, 15(23):9070.

Eschenbaum, N., Mellgren, F., and Zahn, P. (2022). Robust algorithmic collusion. *arXiv preprint arXiv:2201.00345*.

Feng, C. and Liu, A. L. (2025). Peer-to-peer energy trading of solar and energy storage: A networked multiagent reinforcement learning approach. *Applied Energy*, 383:125283.

Gao, Y., Wang, W., and Yu, N. (2021). Consensus multi-agent reinforcement learning for volt-var control in power distribution networks. *IEEE Transactions on Smart Grid*, 12(4):3594–3604.

Guerrero, J., Gebbran, D., Mhanna, S., Chapman, A. C., and Verbič, G. (2020). Towards a transactive energy system for integration of distributed energy resources: Home energy management, distributed optimal power flow, and peer-to-peer energy trading. *Renewable and Sustainable Energy Reviews*, 132:110000.

Hönen, J., Hurink, J. L., and Zwart, B. (2023a). A classification scheme for local energy trading. *OR Spectrum*, 45(1):85–118.

Hönen, J., Hurink, J. L., and Zwart, B. (2023b). Dynamic rolling horizon-based robust energy management for microgrids under uncertainty. *arXiv preprint arXiv:2307.05154*.

Hu, J. and Wellman, M. P. (2003). Nash q-learning for general-sum stochastic games. *Journal of machine learning research*, 4(Nov):1039–1069.

Jensen, A., Anderson, K., Holmgren, W., Mikofski, M., Hansen, C., Boeman, L., and Loonen, R. (2023). pvlib iotools — open-source python functions for seamless access to solar irradiance data. *Solar Energy*, 266:112092.

Khorasany, M., Mishra, Y., and Ledwich, G. (2018). Market framework for local energy trading: A review of potential designs and market clearing approaches. *IET Generation, Transmission & Distribution*, 12(22):5899–5908.

Kloppenburg, S. and Boekelo, M. (2019). Digital platforms and the future of energy provisioning: Promises and perils for the next phase of the energy transition. *Energy Research & Social Science*, 49:68–73.

Krabichler, T. and Teichmann, J. (2023). A case study for unlocking the potential of deep learning in asset-liability-management. *Frontiers in Artificial Intelligence*, 6:1177702.

Lara, J. D., Olivares, D. E., and Canizares, C. A. (2018). Robust energy management of isolated microgrids. *IEEE Systems Journal*, 13(1):680–691.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D. (2015). Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*.

Littman, M. L. (1994). Markov games as a framework for multi-agent reinforcement learning. In *Machine learning proceedings 1994*, pages 157–163. Elsevier.

Littman, M. L. et al. (2001). Friend-or-foe q-learning in general-sum games. In *ICML*, volume 1, pages 322–328.

Long, C., Wu, J., Zhang, C., Thomas, L., Cheng, M., and Jenkins, N. (2017). Peer-to-peer energy trading in a community microgrid. In *2017 IEEE power & energy society general meeting*, pages 1–5. IEEE.

Lowe, R., Wu, Y. I., Tamar, A., Harb, J., Pieter Abbeel, O., and Mordatch, I. (2017). Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in neural information processing systems*, 30.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533.

Qiu, D., Wang, J., Wang, J., and Strbac, G. (2021). Multi-agent reinforcement learning for automated peer-to-peer energy trading in double-side auction market. In *IJCAI*, pages 2913–2920.

Rodrigues, D. L., Ye, X., Xia, X., and Zhu, B. (2020). Battery energy storage sizing optimisation for different ownership structures in a peer-to-peer energy sharing community. *Applied Energy*, 262:114498.

Roesch, M., Linder, C., Zimmermann, R., Rudolf, A., Hohmann, A., and Reinhart, G. (2020). Smart grid for industry using multi-agent reinforcement learning. *Applied Sciences*, 10(19):6900.

Rosen, C. and Madlener, R. (2016). Regulatory options for local reserve energy markets: Implications for prosumers, utilities, and other stakeholders. *The Energy Journal*, 37(2_suppl):39–50.

Samadi, E., Badri, A., and Ebrahimpour, R. (2020). Decentralized multi-agent based energy management of microgrid using reinforcement learning. *International Journal of Electrical Power & Energy Systems*, 122:106211.

Samende, C., Cao, J., and Fan, Z. (2022). Multi-agent deep deterministic policy gradient algorithm for peer-to-peer energy trading considering distribution network constraints. *Applied Energy*, 317:119123.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. (2016). Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489.

Sousa, T., Soares, T., Pinson, P., Moret, F., Baroche, T., and Sorin, E. (2019). Peer-to-peer and community-based markets: A comprehensive review. *Renewable and Sustainable Energy Reviews*, 104:367–378.

Tsaousoglou, G., Giraldo, J. S., and Paterakis, N. G. (2022). Market mechanisms for local electricity markets: A review of models, solution concepts and algorithmic techniques. *Renewable and Sustainable Energy Reviews*, 156:111890.

Tushar, W., Nizami, S., Azim, M. I., Yuen, C., Smith, D. B., Saha, T., Poor, H. V., et al. (2023). Peer-to-peer energy sharing: A comprehensive review. *Foundations and Trends® in Electric Energy Systems*, 6(1):1–82.

Tushar, W., Yuen, C., Saha, T. K., Morstyn, T., Chapman, A. C., Alam, M. J. E., Hanif, S., and Poor, H. V. (2021). Peer-to-peer energy systems for connected communities: A review of recent advances and emerging challenges. *Applied energy*, 282:116131.

Weiller, C. M. and Pollitt, M. G. (2013). *Platform markets and energy services*. JSTOR.

Zhang, K., Yang, Z., and Basar, T. (2018a). Networked multi-agent reinforcement learning in continuous spaces. In *2018 IEEE conference on decision and control (CDC)*, pages 2771–2776. IEEE.

Zhang, K., Yang, Z., and Başar, T. (2021). Multi-agent reinforcement learning: A selective overview of theories and algorithms. *Handbook of reinforcement learning and control*, pages 321–384.

Zhang, K., Yang, Z., Liu, H., Zhang, T., and Basar, T. (2018b). Fully decentralized multi-agent reinforcement learning with networked agents. In *International conference on machine learning*, pages 5872–5881. PMLR.

Zhou, Y., Wu, J., Long, C., and Ming, W. (2020). State-of-the-art analysis and perspectives for peer-to-peer energy trading. engineering 6 (7): 739–753.

A Existence of a global minimum for the centralized optimizer

We first state the formal assumptions on the model on the admissible prices (Assumption 1) and on the admissible actions and fees (Assumption 2). We then provide a simple formal proof for the existence of a cost-minimum (or social planner optimum) in Proposition 1.

Assumption 1 (Admissible Prices)**.** *The platform price functions*

$$p_t^D, p_t^S : \mathbb{R}_+^2 \to [p_{\text{fit}},\, p^I] \tag{10}$$

for each period $t = 1, \ldots, T$ satisfy:

(i) ***Continuity:*** *p_t^D, p_t^S are continuous in (D^t, S^t).*

(ii) ***Weak monotonicity:***

$$\frac{\partial p_t^D}{\partial D^t} \geq 0, \quad \frac{\partial p_t^D}{\partial S^t} \leq 0, \quad \frac{\partial p_t^S}{\partial S^t} \leq 0, \quad \frac{\partial p_t^S}{\partial D^t} \geq 0. \tag{11}$$

(iii) ***Price corridor:***

$$p_{\text{fit}} \;\leq\; p_t^S(D^t, S^t) \;\leq\; p_t^D(D^t, S^t) \;\leq\; p^I. \tag{12}$$

(iv) ***Budget balance:***

$$\begin{cases} D^t \geq S^t: & (D^t - S^t)\, p^I = D^t\, p_t^D(D^t, S^t) \;-\; S^t\, p_t^S(D^t, S^t), \\ beginequation4pt]S^t > D^t: & (S^t - D^t)\, p_{\text{fit}} = S^t\, p_t^S(D^t, S^t) \;-\; D^t\, p_t^D(D^t, S^t). \end{cases} \tag{13}$$

Assumption 2 (Technological Constraints)**.** *There are $N < \infty$ agents. Each agent i chooses $\{(b_t^i, \kappa_t^i)\}_{t=1}^T$ in the compact set*

$$A_T^i = \prod_{t=1}^{T} \left[-B_{t-1}^i,\, \overline{B}^i - B_{t-1}^i \right] \times [0, 1], \tag{14}$$

where B_{t-1}^i is the state-of-charge at $t-1$, $\overline{B}^i$ the capacity. Fixed fees $\{\tau_t^c, \tau_t^p\}$ are nonnegative and exogenous.

Proposition 1 (Existence of Social Planner Optimum)**.** *Under Assumptions 1 and 2, the social planner's problem*

$$\min_{\{b_t^i,\kappa_t^i\}} \sum_{i=1}^{N}\sum_{t=1}^{T} \begin{cases} x_t^i p_t^D(D^t,S^t), & x_t^i \geq 0, \\ [4pt] x_t^i p_t^S(D^t,S^t), & x_t^i < 0, \end{cases} \tag{15}$$

where $x_t^i = d_t^i + b_t^i - \kappa_t^i g_t^i$, $D^t = \sum_i \max\{x_t^i, 0\}$, $S^t = \sum_i \max\{-x_t^i, 0\}$, *admits at least one global minimiser.*

Proof. The joint action space $\mathcal{F} = \prod_{i=1}^{N} A_T^i$ is a finite product of compact intervals, hence compact. The maps $(b_t^i, \kappa_t^i) \mapsto (x_t^i) \mapsto (D^t, S^t) \mapsto (p_t^D, p_t^S)$ are continuous by Assumption 1. The stage-cost $x_t^i \mapsto x_t^i\, p$ is continuous and matches at $x_t^i = 0$. Therefore the total cost is continuous on $\mathcal{F}$, and by Weierstrass's theorem it attains a minimum. □