SAFE MULTI-AGENT REINFORCEMENT LEARNING FOR PRICE-BASED DEMAND RESPONSE

Hannah Markgraf and Matthias Althoff
Cyber-Physical Systems Group
Technical University of Munich
Garching b. München, Germany
{mhan, althoff}@in.tum.de

ABSTRACT

Price-based demand response (DR) enables households to provide the flexibility required in power grids with a high share of volatile renewable energy sources. Multi-agent reinforcement learning (MARL) offers a powerful, decentralized decision-making tool for autonomous agents participating in DR programs. Unfortunately, MARL algorithms do not naturally allow one to incorporate safety guarantees, preventing their real-world deployment. To meet safety constraints, we propose a safety layer that minimally adjusts each agent's decisions. We investigate the influence of using a reward function that reflects these safety adjustments. Results show that considering safety aspects in the reward during training improves both convergence speed and performance of the MARL agents in the investigated numerical experiments.

1 INTRODUCTION

The electrification of the heating and mobility sector entails both challenges and opportunities for power grid operation. Controllable electric loads can provide a source of flexibility for meeting the volatility in renewable generation, but coordination is required to take full advantage of this flexibility and avoid undesired demand peaks. Mechanisms to adjust the power consumption patterns of end customers for improving the reliability of power grid operation and decreasing operational costs are summarized under the term demand response (DR). For private households, DR mechanisms are usually price-based, meaning that electricity pricing schemes are used to incentivize a desired consumption behavior of customers acting as autonomous agents (Siano, 2014).

Early research in this field focused on the day-ahead consumption scheduling of autonomous agents under dynamic pricing schemes (Mohsenian-Rad et al., 2010; Li et al., 2011), neglecting uncertainties in both consumer behavior and generation forecasting. For training autonomous agents in real-time DR settings, multi-agent reinforcement learning (MARL) offers a model-free and adaptive framework with the capability of handling large amounts of data as well as complex nonlinear problems (Vázquez-Canteli & Nagy, 2019). MARL has been applied successfully for reducing peak loads (Vázquez-Canteli et al., 2020; Ebell & Pruckner, 2021; Bahrami et al., 2021) as well as operational costs (Christensen et al., 2020; Shojaeighadikolaei et al., 2021). Despite these promising results, practical deployment of MARL for DR is hindered by its inability to incorporate physical constraints such as storage capacities. Existing work addresses this problem using reward shaping (Kofinas et al., 2018), over-dimensioning components (Vázquez-Canteli et al., 2020), or employing backup controllers on the component level (Vázquez-Canteli et al., 2020; Bahrami et al., 2021). Of these options, only backup controllers can guarantee constraint satisfaction. However, previous work fails to investigate the influence of overriding decisions of the MARL algorithm.

We address this research gap by incorporating a constraint violation penalty into the reward used during MARL training. Our proposed safety layer ensures minimal interference by using a shrinking horizon model-predictive control formulation (Ye & Kolmanovsky, 2022) to project each agent's decision onto the feasible set defined by agent-specific constraints. The penalty is proportional to the necessary adjustment. We compare training with and without the penalty added to the reward and

use a perfect-knowledge day-ahead scheduling approach (Li et al., 2011) as a theoretical maximum to evaluate overall performance of the real-time MARL approach.

2 PROBLEM FORMULATION

We consider a group of N households served by a single energy provider. Each household participates in the DR program to optimize its own payoff. The goal of the energy provider is to minimize the operational cost as well as the aggregated discomfort. Since it cannot directly manage the households' power consumption, it issues dynamic prices to incentivize the households to adopt a socially optimal behavior. Next, we describe the case study and the perfect-knowledge solution approach from Li et al. (2011), which we use as a theoretical maximum for algorithm performance. Then, we formulate the real-time interactions as a partially observable Markov Game (POMG) to solve it using MARL.

2.1 CASE STUDY

Each household $i \in \{1, ..., N\}$ has a non-shiftable base load $\ell_{i,t}$ corresponding to household appliances (such as dishwashers), a battery, and a shiftable load corresponding to a heating, ventilation, and cooling system (HVAC). We consider a discrete-time setting with time interval Δt and time horizon T. At each time step t, households can shape their power consumption by adjusting the battery charge/discharge power $p^B_{i,t}$ and the power consumed by the HVAC, denoted $p^{AC}_{i,t}$. The overall demand $d_{i,t}$ of a household is obtained using the power balance equation

$$d_{i,t} = \ell_{i,t} + p^{AC}_{i,t} + p^B_{i,t}. \tag{1}$$

The energy stored in the battery, denoted $x^B_{i,t}$, and the indoor temperature $x^{AC}_{i,t}$ evolve according to the discrete-time dynamics adapted from Li et al. (2011, Eq. 1, Eq. 13):

$$x^B_{i,t+1} = x^B_{i,t} + p^B_{i,t}\Delta t, \tag{2}$$

$$x^{AC}_{i,t+1} = x^{AC}_{i,t} + (\alpha_i(T^A_{i,t} - x^{AC}_{i,t}) + \beta_i p^{AC}_{i,t})\Delta t. \tag{3}$$

Here, α_i is a positive constant specifying the building insulation and $T^A_{i,t}$ is the outdoor temperature. Since we only consider the heating scenario, β_i is also a positive constant capturing the thermal characteristics of the HVAC.

We refer to the discomfort $U_{i,t}$ of a household as the sum of the deviation from the desired room temperature T^D_i and a term accounting for the cost of battery usage, such that

$$U_{i,t} = \theta_i(x^{AC}_{i,t} - T^D_i)^2 + \sigma_i(p^B_{i,t}\Delta t)^2. \tag{4}$$

Herein, the sensitivity for temperature deviation is captured by $\theta_i \in (0, 1]$, while σ_i models the equipment cost and expected degradation of the battery. The electricity cost $P_{i,t}$ of a household depends on the electricity price ϕ_t set by the energy provider:

$$P_{i,t} = \phi_t d_{i,t}\Delta t. \tag{5}$$

The goal of the customer is to optimize its payoff over a certain time horizon, e.g., one day, by adjusting its power consumption $p_{i,t} = \begin{bmatrix} p^B_{i,t} & p^{AC}_{i,t} \end{bmatrix}^T$ in each time step $t \in \{1, ..., T\}$. We denote the sequence of power set points over the horizon as $p_{i,(\cdot)}$. Both the power set points and the state variables $x^B_{i,t}$ and $x^{AC}_{i,t}$ are constrained by lower and upper limits, which we refer to as $z \in [\underline{z}, \overline{z}]$ for some quantity z. Furthermore, we set the lower limit of the total demand of a household to 0 to prevent exporting battery power to the grid. Finally, we impose a minimum terminal charge for the

battery. The day-ahead scheduling for one household thus results in solving

$$\min_{p_{i,(\cdot)}} \quad \sum_t U_{i,t} + \sum_t P_{i,t} \tag{6a}$$

$$\text{s.t. } \forall t: \quad p^B_{i,t} \in [\underline{p}^B_i, \overline{p}^B_i], \tag{6b}$$

$$p^{AC}_{i,t} \in [0, \overline{p}^{AC}_i], \tag{6c}$$

$$x^B_{i,t} \in [\underline{x}^B_i, \overline{x}^B_i], \tag{6d}$$

$$x^{AC}_{i,t} \in [\underline{x}^{AC}_i, \overline{x}^{AC}_i], \tag{6e}$$

$$d_{i,t} \in [0, \overline{d}_i], \tag{6f}$$

$$\epsilon_i \overline{x}^B_i \leq x^B_{i,T}, \epsilon_i \in (0, 1]. \tag{6g}$$

Note that the optimal solution of (6a) depends on the sequence of prices $\phi_{(\cdot)}$ set by the energy provider. Next, we detail how to use dynamic pricing to incentivize socially optimal behavior of the households.

2.2 Dynamic pricing mechanism and optimal solution

The social cost of a community of households trades off the discomfort of each individual household with the operational cost of satisfying the aggregated demand $d_t := \sum_i d_{i,t}$. For simplicity, we assume a quadratic operational cost function $C(d_t)$ (Forouzandehmehr et al., 2015). Computing the optimal power consumption of all households $p = \begin{bmatrix} p_{1,(\cdot)} & \dots & p_{N,(\cdot)} \end{bmatrix}^T$ by solving

$$\min_{p} \quad \sum_i \sum_t U_{i,t} + \sum_t C(d_t) \tag{7}$$

$$\text{s.t. } \forall i, \forall t: \quad \text{(6b)} - \text{(6g)}$$

is in theory possible, since both the objective function and the feasible set are convex (Li et al., 2011). However, it would require knowledge of all utilities and constraints. As an alternative, the work in Li et al. (2011) proposes adopting a pricing scheme using

$$\phi_t = \partial C(d_t) / \partial d_t. \tag{8}$$

The authors prove that using an iterative algorithm, an equilibrium between the prices and the power consumption schedules can be reached, which optimizes both the social cost (7) and the aggregation of individual household costs (6a). For more details on the iterative algorithm used to solve this benchmark problem, the reader is referred to (Li et al., 2011).

Computing the equilibrium is well suited for day-ahead scheduling, but would be computationally prohibitive when used in a real-time setting. Furthermore, the approach in (Li et al., 2011) assumes perfect knowledge of the values for $\ell_{i,t}$ and $T^A_{i,t}$. To provide a more realistic solution using MARL, we subsequently formulate the interaction between the households and the energy provider as a POMG.

3 Safe MARL for Demand Response

A POMG models the interaction between a set of agents $i \in \mathcal{N}$ and an environment as a sequential decision-making process and thereby provides a theoretical framework for MARL. It consists of a tuple $(\mathcal{N}, \mathcal{S}, \{\mathcal{A}_i, \mathcal{O}_i, R_i\}_{\forall i})$, where the notation $\{\}_{\forall i}$ refers to the collection of individual quantities (Gronauer & Diepold, 2022). At each time step t, each agent (corresponding to one household) receives an observation $o_{i,t} = \begin{bmatrix} x^B_{i,t} & x^{AC}_{i,t} & T^A_{i,t} & T^D_i & \ell_{i,t} & d_{i,t} & \phi_t & h \end{bmatrix}^T \in \mathcal{O}_i$, where h refers to the hour of the day. Based on these individual observations, the agents select an action $a_{i,t} = \begin{bmatrix} p^B_{i,t} & p^{AC}_{i,t} \end{bmatrix}^T \in \mathcal{A}_i$. This action is applied to the environment, whose global state $s_t = \begin{bmatrix} \{x^B_{i,t} & x^{AC}_{i,t} & T^A_{i,t} & T^D_i & \ell_{i,t} & d_{i,t}\}_{\forall i} & \phi_t & h \end{bmatrix}^T \in \mathcal{S}$ transitions to the next state. The transitions of the state variables $x^B_{i,t}$ and $x^{AC}_{i,t}$ are determined by (2) and (3). The total load $d_{i,t}$ and the price ϕ_t are computed by (1) and (8). The environment issues agent-specific rewards

$$R_{i,t} = -(U_{i,t} + P_{i,t}). \tag{9}$$

The negation is used because the goal of each agent is to maximize its expected cumulative reward, not minimize its cost. We reset the environment at the end of a day ($t = T$), such that the goal of each RL agent is aligned with the objective in (6a).

RL algorithms do not naturally allow one to incorporate constraints such as (6b) - (6g). Therefore, we use a safety layer based on a shrinking horizon MPC formulation (SHMPC) (Ye & Kolmanovsky, 2022). As opposed to receding horizon MPC, where a control problem is solved repeatedly over a moving time horizon with a fixed length, the SHMPC solves the problem over a shrinking horizon between the current time step t and the fixed final time step T. Hence, the number of optimization variables is reduced with each time step, which benefits computation time. However, our safety layer can easily be reformulated for a receding horizon MPC.

Since the constraints in (6b) - (6g) are not coupled between the agents, we can formulate an individual safety layer for every agent. At each time step t , this safety layer solves a constrained optimization problem to adjust the action $a_{i,t}$ proposed by the RL agent while ensuring that the constraints are satisfied for all remaining time steps $\tau \in \{t, ..., T\}$. To remove the perfect-knowledge assumption in Li et al. (2011), we use predictions $\hat{T}^A_{i,\tau|t}$ and $\hat{\ell}^{AC}_{i,\tau|t}$ at some point τ in the future based on the knowledge at time point t. Consequently, we have to adjust (1) and (3) such that

$$x^{AC}_{i,\tau+1|t} = x^{AC}_{i,\tau|t} + (\alpha_i(\hat{T}^A_{i,\tau|t} - x^{AC}_{i,\tau|t}) + \beta_i p^{AC}_{i,\tau})\Delta t \tag{10}$$

$$d_{i,\tau|t} = \hat{\ell}_{i,\tau|t} + p^{AC}_{i,\tau} + p^{B}_{i,\tau}. \tag{11}$$

We expect predictions for some quantity z as $\hat{z}_{\tau|t} = z_\tau + \xi_{\tau|t}$, where z_τ corresponds to the true value. The noise $\xi_{\tau|t}$ is uniformly sampled from an interval $\Xi_{\tau|t} = [\underline{\xi}_{\tau|t}, \overline{\xi}_{\tau|t}]$, the size of which linearly increases over the horizon (Pinson & Kariniotakis, 2004). This implies that, given a prediction, the true value of the quantity will lie within $z_\tau \in [\hat{z}_{\tau|t} - \overline{\xi}_{\tau|t}, \hat{z}_{\tau|t} - \underline{\xi}_{\tau|t}]$. Using these worst-case bounds, the resulting optimization problem solved by the safety layer is

$$\min_{\tilde{a}_{i,(\cdot)}} \quad \|a_{i,t} - \tilde{a}_{i,t}\|_2 \tag{12a}$$

$$\text{s.t. } \forall \tau: \quad p^B_{i,\tau} \in [\underline{p}^B_i, \overline{p}^B_i], \tag{12b}$$

$$p^{AC}_{i,\tau} \in [0, \overline{p}^{AC}_i], \tag{12c}$$

$$x^B_{i,\tau} \in [\underline{x}^B_i, \overline{x}^B_i], \tag{12d}$$

$$x^{AC}_{i,\tau|t} \in [\underline{x}^{AC}_i + \alpha_i \Delta t \overline{\xi}_{\tau|t}, \overline{x}^{AC}_i + \alpha_i \Delta t \underline{\xi}_{\tau|t}] \tag{12e}$$

$$d_{i,\tau|t} \in [\overline{\xi}_{\tau|t}, \overline{d}_i + \underline{\xi}_{\tau|t}] \tag{12f}$$

$$\epsilon_i \overline{x}^B_i \leq x^B_{i,T}, \epsilon_i \in (0, 1]. \tag{12g}$$

The constraints formulated in (12b) - (12d) and (12g) are the same as in (6b) - (6d) and (6g), respectively. (12e) and (12f) specify the worst-case bounds for the temperature and the total load, which are computed using the predicted quantities as in (10) and (11). The objective function (12a) minimizes the distance between the safe action $\tilde{a}_{i,t}$ and the action suggested by the RL agent. Note that even though we optimize over the actions for all remaining time steps, $\tilde{a}_{i,(\cdot)}$, we are only interested in obtaining the safe action for the current time step, $\tilde{a}_{i,t}$. This action is then passed on to the environment.

If an action correction is necessary, we compute a penalty $\tilde{W}_{i,t} = c\|a_{i,t} - \tilde{a}_{i,t}\|_2$, where c is a positive constant. To incentivize the agent to minimize the constraint violation, we add this penalty to the reward, such that

$$\tilde{R}_{i,t} = -(\tilde{U}_{i,t} + \tilde{P}_{i,t} + \tilde{W}_{i,t}), \tag{13}$$

where $\tilde{U}_{i,t}$ and $\tilde{P}_{i,t}$ correspond to the discomfort and electricity cost for the safe action.

To solve the POMG, we employ Multi-agent Proximal Policy Optimization (MAPPO) (Yu et al., 2022), an actor-critic MARL algorithm. The actor-critic architecture is particularly suitable for addressing the non-stationarity (Gronauer & Diepold, 2022) introduced through the dynamic pricing: Since the critic is only used during training, it allows one to make additional information available during training (e.g., the global state), which is then removed during execution time. This training paradigm is referred to as centralized training, decentralized execution (CTDE). For further information, the reader is referred to Gronauer & Diepold (2022).

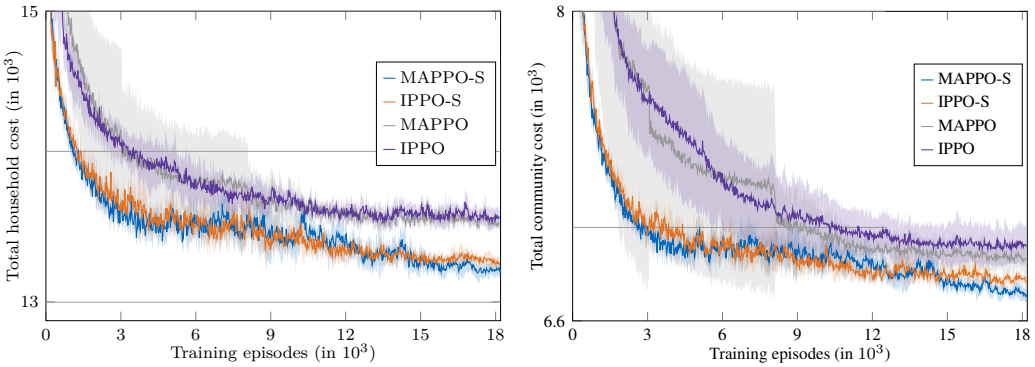

Figure 1: Evolution of aggregated household cost (left) and social cost (right) during training.

4 Experiments and Discussion

We train five agents with individual system parameters given in Appendix A.1 to schedule the power consumption on a given day in January. We compare the CTDE paradigm to independent learning (IPPO), where no global state information is used during training. To investigate the influence of the constraint violation penalty, we compare the training and testing performance when using the reward functions $R_{i,t}$ and $\tilde{R}_{i,t}$, respectively. We denote training runs with reward $\tilde{R}_{i,t}$ as MAPPO-S and IPPO-S. For MAPPO, IPPO, MAPPO-S, and IPPO-S, six training runs with different random seeds are conducted. The hyperparameters we use are specified in Appendix A.3.

The training curves shown in Fig. 1 indicate that using $\tilde{R}_{i,t}$ during training improves both performance and convergence speed. During deployment, the MAPPO-S and IPPO-S agents not only achieve lower costs, but also require significantly fewer interference of the safety layer than their counterparts trained with $R_{i,t}$, as can be seen in Tab. 1. This is a crucial result, since existing approaches use MARL for DR without any consideration of safety aspects in the reward. Furthermore, Tab. 1 shows that the MAPPO-S agents achieve a substantially lower peak consumption, which could be attributed to the global state information used during training.

5 Conclusion and Future Work

We present a safe MARL algorithm for price-based DR, facilitating the deployment in real-world settings. The proposed safety layer ensures constraint satisfaction by projecting each agent's action into the agent-specific feasible set. We show that training the agents with the reward that accounts for the constraint violation speeds up convergence and yields better overall results with respect to both aggregated individual and social costs. Using this reward, the MAPPO-S algorithm approaches the optimal day-ahead solution. Future work should integrate forecasts of observations and prices to further improve the MARL performance. Futhermore, we will focus on handling constraints at community level, e.g., limiting the aggregated demand of all households to the maximum transformer power.

Table 1: Benchmarking deployment of best models obtained with different MARL training schemes.

	Optimum	IPPO-S	IPPO	MAPPO-S	MAPPO
Community cost	6,564.45	6,703.72	6,797.82	6,642.63	6,799.45
Total household cost	12,966.36	13,109.11	13,348.47	13,088.02	13,453.85
Peak load	18.36	22.5	26.43	20.89	29.12
Action corrections	-	98	408	48	446

ACKNOWLEDGMENTS

This work was supported by the Bavarian Research Foundation project STROM (Energy - Sector coupling and microgrids).

REFERENCES

Shahab Bahrami, Yu Christine Chen, and Vincent W. S. Wong. Deep Reinforcement Learning for Demand Response in Distribution Networks. *IEEE Transactions on Smart Grid*, 12(2):1496–1506, 2021.

Morten Herget Christensen, Cédric Ernewein, and Pierre Pinson. Demand Response through Price-setting Multi-agent Reinforcement Learning. In *Proceedings of the 1st International Workshop on Reinforcement Learning for Energy Management in Buildings & Cities*, pp. 1–5, 2020.

Niklas Ebell and Marco Pruckner. Benchmarking a Decentralized Reinforcement Learning Control Strategy for an Energy Community. In *IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, pp. 385–390, 2021.

Najmeh Forouzandehmehr, Mohammad Esmalifalak, Hamed Mohsenian-Rad, and Zhu Han. Autonomous Demand Response Using Stochastic Differential Games. *IEEE Transactions on Smart Grid*, 6(1):291–300, 2015.

Sven Gronauer and Klaus Diepold. Multi-agent deep reinforcement learning: a survey. *Artificial Intelligence Review*, 55(2):895–943, 2022.

Panagiotis Kofinas, Anastasios I. Dounis, and George A. Vouros. Fuzzy Q-Learning for multi-agent decentralized energy management in microgrids. *Applied Energy*, 219:53–67, 2018.

Na Li, Lijun Chen, and Steven H. Low. Optimal demand response based on utility maximization in power networks. In *IEEE Power and Energy Society General Meeting*, pp. 1–8, 2011.

Amir-Hamed Mohsenian-Rad, Vincent W. S. Wong, Juri Jatskevich, Robert Schober, and Alberto Leon-Garcia. Autonomous Demand-Side Management Based on Game-Theoretic Energy Consumption Scheduling for the Future Smart Grid. *IEEE Transactions on Smart Grid*, 1(3):320–331, 2010.

Pierre Pinson and Georges Kariniotakis. On-line assessment of prediction risk for wind power production forecasts. *Wind Energy*, 7(2):119–132, 2004.

Amin Shojaeighadikolaei, Arman Ghasemi, Kailani R. Jones, Alexandru G. Bardas, Morteza Hashemi, and Reza Ahmadi. Demand Responsive Dynamic Pricing Framework for Prosumer Dominated Microgrids using Multiagent Reinforcement Learning. In *52nd North American Power Symposium (NAPS)*, 2021.

Pierluigi Siano. Demand response and smart grids—A survey. *Renewable and Sustainable Energy Reviews*, 30:461–478, 2014.

José R. Vázquez-Canteli and Zoltán Nagy. Reinforcement learning for demand response: A review of algorithms and modeling techniques. *Applied Energy*, 235:1072–1089, 2019.

José R. Vázquez-Canteli, Gregor Henze, and Zoltan Nagy. MARLISA: Multi-Agent Reinforcement Learning with Iterative Sequential Action Selection for Load Shaping of Grid-Interactive Connected Buildings. In *Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, pp. 170–179, 2020.

Mingfei Ye and Ilya Kolmanovsky. Approximating optimal control by shrinking horizon model predictive control for spacecraft rendezvous and docking. *IFAC-PapersOnLine*, 55(16):284–289, January 2022. URL https://www.sciencedirect.com/science/article/pii/S2405896322012149.

Chao Yu, Akash Velu, Eugene Vinitsky, Yu Wang, Alexandre Bayen, and Yi Wu. The Surprising Effectiveness of PPO in Cooperative, Multi-Agent Games, 2022. URL https://arxiv.org/abs/2103.01955.

A PARAMETER VALUES

A.1 SYSTEM PARAMETERS

Table 2: System parameters for five households.

	Household 1	Household 2	Household 3	Household 4	Household 5
α_i	0.9	0.9	0.9	0.9	0.9
β_i	7.61	6.09	6.47	7.5	6.9
ϵ_i	0.5	0.5	0.5	0.5	0.5
σ_i	10.0	10.0	10.0	10.0	10.0
θ_i	1.0	1.0	1.0	1.0	1.0
T_i^D [°C]	24	24	23	22	23
$\overline{p}_i^B$ [kW]	1.8	1.8	1.8	1.8	1.8
$\overline{p}_i^{AC}$ [kW]	4	4	4	4	4
$\underline{x}_i^B$ [kWh]	0.325	0.325	0.325	0.325	0.325
$\overline{x}_i^B$ [kWh]	6.175	6.175	6.175	6.175	6.175
$\underline{x}_i^{AC}$ [°C]	20	20	20	20	20
$\overline{x}_i^{AC}$ [°C]	26	26	26	26	26
$\overline{d}_i$ [kW]	10	10	10	10	10
c	1	1	1	1	1

A.2 PREDICTIONS AND WORST-CASE BOUNDS

We generate the predictions $\hat{T}^A_{i,\tau|t}$ and $\hat{\ell}_{i,\tau|t}$ by adding smoothed uniform noise to the true values. The base amplitude of the uniform noise is 0.05 and increases linearly with a factor of 1.08 over the horizon. To smooth the noise, we apply two iterations of a moving-average filter with window size 19.

A.3 HYPERPARAMETERS

We employ the standard hyperparameters [1] except for the values listed in Tab. 3.

Table 3: Hyperparameters for MAPPO and IPPO.

Parameter	Explanation	Value
clip_param	Clipping threshold for policy updates	0.1
critic_lr	Learning rate for critic network	0.0003
lr	Learning rate for policy network	0.0003
ppo_epoch	Number of gradient step updates per rollout	5

[1] provided in https://github.com/marlbenchmark/on-policy