

Multi-Objective Reinforcement Learning with Shielding for Control and Safety in Power Grids

GÖKHAN DEMIREL, SIMON GRAFENHORST, JAKOB OHM, BENJAMIN SCHÄFER, KEVIN FÖRDERER, and VEIT HAGENMEYER, Karlsruhe Institute of Technology, Germany

Traditional model-based methods relying on precise modeling are sensitive to errors and rapidly changing conditions, causing unstable control. Furthermore, real-world problems often involve conflicting objectives, requiring multi-objective reinforcement learning (MORL) in order to facilitate adaptability. However, MORL and single-objective RL (SORL) research often address these challenges separately due to the lack of a unified framework. To bridge this gap, we present eGridLVGym, an open-source collection of two test environments for managing low-voltage grids. The first environment models a household with rooftop photovoltaics, battery storage, electric vehicle charging, and a heat pump. The second extends this to a feeder grid with seven points of common coupling, representing diverse portfolios of distributed energy resources. These environments support system-wide flexibility dispatch while ensuring safety through soft and hard constraints that penalize or restrict unsafe actions. We evaluate eight state-of-the-art SORL algorithms (A2C, DDPG, PPO, SAC, TD3, Recurrent PPO, TQC, and TRPO), a grid code shielding approach, and the MORL algorithm CAPQL. Our results demonstrate that the best SORL algorithm reduces transformer overload from 93% to 75% and cuts economic costs to 122%, while the default-weighted MORL outperforms the rule-based controller by 77% in economic cost.

Additional Key Words and Phrases: Multi-objective Reinforcement Learning, Energy Management, Voltage Control

1 Introduction

Global energy systems are rapidly evolving towards many possible future architectures, driven by a transition to renewable energy sources (RES) and distributed energy resources (DER). As integrating renewables leads to increased volatility and unpredictable events due to their inherent variability, the corresponding tighter supply constraints increase the complexity of operational and planning decisions in energy systems [42]. Consequently, controllable loads, such as heat pumps (HP), and other DER, such as battery energy storage (BES), will become increasingly important for providing flexibility and thereby raising the resilience of the energy system in order to deal with the uncertainties of nature. To tackle these challenges, deep reinforcement learning (RL) offers a powerful solution, proven in domains like games (e.g., Atari [44], Go [37]) and robotics [23]. These advancements provide researchers with tools to address the complexity of modern energy systems.

Deep decarbonization pathways, as in the New York State, require multidimensional flexibility for reliability and cost balance [10]. On the one hand, implicit flexibility arises from prosumers actively adapting their energy consumption in response to price signals triggered by grid tariffs. On the other hand, explicit flexibility involves

the trading of dispatchable resources on the energy markets to assist Distribution System Operators (DSO) in managing grid congestion and reducing costs for ancillary services [26].

The complexity of non-linear energy systems coupled with the increasing integration of DER and RES, dynamic tariffs, and hard constraints make conventional control approaches inefficient, as their computational burden increases exponentially with the size of the conflicting objectives. RL offers a promising potential for addressing the automation of the arising complex sequential decision-making problems in energy [4, 6]. To minimize the risks of unsafe actions and ensure safe operation in our present work, the safe single-objective RL (SORL) techniques [39, 47] and multi-objective RL (MORL) shape the interaction of agents with the environment embedded in a multi-objective markov decision process (MOMDP) [34]. This approach minimizes costs, including energy losses, through joint strategies, for instance controlling battery electric vehicle (BEV) while maintaining voltage stability. Despite advances in RL, existing methods often focus on optimizing single objectives or ignore safety and regulatory requirements, limiting their practical applicability in real-world energy systems. In addition, the comparison of aggregated single-objective and multi-objective approaches to multi-objective problems has not been sufficiently explored. To address these gaps, the following research questions drive the present paper:

- (1) How can RL tackle multi-objective tasks for DERs (e.g., BES, HP, and EVs) in future energy systems?
- (2) How do state-of-the-art SORL algorithms (A2C, DDPG, PPO, SAC, TD3, Recurrent PPO, TQC, and TRPO) perform when handling multi-objective tasks with scalarized rewards?
- (3) Can grid code-based shielding mechanisms ensure safety and regulatory compliance in RL-based energy control?
- (4) How do MORL algorithms compare to SORL in terms of performance and adaptability?

Figure 1 illustrates the proposed reinforcement learning environment, comprising single-objective, multi-objective, and shielded

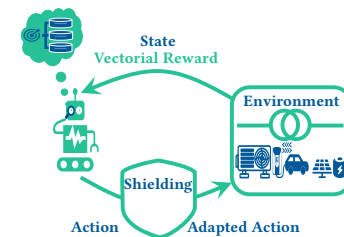


Fig. 1. A reinforcement learning framework for single-, multi-objective-, and shielded control in future energy systems.

Authors' address: Gökhan Demirel, goekhan.demirel@kit.edu; Simon Grafenhorst, simon.grafenhorst@kit.edu; Jakob Ohm, jakob.ohm@student.kit.edu; Benjamin Schäfer, benjamin.schaefer@kit.edu; Kevin Förderer, kevin.foerderer@kit.edu; Veit Hagenmeyer, veit.hagenmeyer@kit.edu, Karlsruhe Institute of Technology, Hermann-von-Helmholtz-Platz 1, Eggenstein-Leopoldshafen, Germany, 76344.

control strategies for heterogeneous DERs. In this paper, we apply SORL and MORL control to heterogeneous DERs. The main contributions of this paper are:

- We conduct extensive numerical experiments with eight state-of-the-art SORL algorithms in multiple seasonal conditions. We integrate these via wrappers that interface with Stable-Baselines3 [33], validating performance in Partially Observable Markov Decision Process (POMDP) for low-voltage (LV) distribution grids.
- We formally define economic, flexibility and grid-oriented problems as a Multi-Objective Partially Observable Markov Decision Process (MOPOMDP) and introduce eGridLVGym, an open-source Python framework.¹ We also provide compatibility with MORL-Baselines [12] for MORL algorithms.
- From the experimental results, we derive challenges (e.g., interpretability) for SORL, SORL via shielding and MORL in distributed energy systems and propose possible directions for future research pathways.

The remainder of this paper is structured as follows: Section 2 shows the related literature on RL for sustainable energy system tasks, including shielding and soft constraint SORL and MORL algorithms for distribution grid control. Section 3 presents the mathematical problem formulations for DERs, the underlying RL environment design, and the associated physical modeling. Section 4 translates this problem into an RL control framework, focusing on SORL and MORL approaches as well as grid code shielding for safe operations. Section 5 describes the setup and presents the main results, while Section 6 summarizes our key findings and outlines future research.

2 Related Work

In the second decade of the 21st century, following breakthroughs in game-based reinforcement learning, RL applications expanded to real-world applications such as energy systems, healthcare, finance, autonomous vehicles, and sustainability.

The Grid2Op [9] environment, maintained by the French Transmission System Operator (TSO) Réseau de Transport d'Electricité (RTE), is an environment for RL research on the management of transmission grids through actions that reconfigure substation nodes to optimize grid performance. The environment provides predefined discrete actions for line management and continuous redispatch of power generation within a Markov Decision Process (MDP). The CityLearn [43] environment uses simplified building models to optimize heating, ventilation, and air-conditioning (HVAC) and BES control to reduce energy consumption and smooth load peaks to minimize the aggregated net energy demand of the district. GridLearn [32] extends the CityLearn framework by embedding building models into a grid model, enabling power flow calculations to address voltage regulation. In contrast, BOPTEST-Gym [3] enables the evaluation of control strategies using highly detailed building models and embedded baseline controllers. The observation space covers the grid topology, power flows, and demand data. GYM-ANM [16] focuses on voltage regulation and works on a six-bus distribution grid. PowerGridworld [2] offers a flexible alternative to CityLearn and GridLearn with configurable time-step resolutions but lacks detailed grid modeling and a baseline controller for

benchmarking. In the latter work, the multi-agent RL (MARL) algorithm operates with heterogeneous agents, setting the rewards for photovoltaic (PV) and BES to zero, focusing solely on HPs. This simplification can disrupt the coherent functioning of MARL, limiting its effectiveness and its broader applicability. Similarly, [45] models only homogeneous DERs, focusing solely on voltage control while neglecting transformer loading, energy pricing, and DER heterogeneity. In [7] a RL agent is trained in the distribution grid to control turbine governors and stabilize frequency after load changes, applying a higher penalty to the final action to ensure a stable state at the end of the episode. Again, no baseline controller is given for comparison; only a Deep Deterministic Policy Gradient (DDPG) algorithm is applied for analysis. BEAR [48], on the other hand, provides an open physics-based platform for building energy management. SustainGym [46] presents five RL environments for sustainable control tasks, including BEV charging scheduling, BES operations, and building controls. EV2Gym [31] provides a benchmarking for by supporting bidirectional charging strategies for RL.

Many of these related energy RL studies address narrow objectives such as voltage regulation [32], HVAC control [43, 48], or frequency stabilization [7] often with homogeneous DERs and scalar rewards. In contrast, our environment aims at multi-objective coordination, involving heterogeneous DERs and conflicting objectives such as voltage stability, transformer loading, flexibility, and cost. Frameworks such as CityLearn or BEAR include DERs but still have a single objective (cost or load shaping). Our contribution closes this gap by evaluating SORL and MORL under grid-code constraints with both scalar and vector rewards. This practice neither decomposes, explicitly scales, nor normalizes individual reward terms to a specific range. Instead, it uses their relative values, resulting in ambiguous weighting factors among components. By contrast, MORL enables the definition of multiple reward functions with conflicting objectives, such as energy consumption and CO₂ emissions, and the identification of trade-offs using Pareto front MORL algorithms. Decomposing scalar rewards for explainable RL (XRL) makes decisions in the XRL domain more explainable and interpretable [21]. Although researchers widely apply RL in energy systems, they have yet to fully explore using MORL algorithms. The lack of a standardized environment for evaluating SORL and MORL algorithms makes comparisons under identical conditions difficult. To fill this gap, we propose eGridLVGym, a Gym-based environment tailored to controlling heterogeneous DERs in LV distribution grids under both scalar and vectorial reward structures. It is designed to control heterogeneous DERs in LV grids and provides valuable insights for researchers and DSOs. We use StableBaselines3 [33] for the SORL algorithms and MORL-baselines [12, 30] for the MORL algorithms to control the DER components. To the best of our knowledge, this is the first unified environment that systematically evaluates both SORL and MORL algorithms on power distribution systems (see Table 1).

3 Background and Design of eGridLVGym

3.1 Problem Formulation in Power Grids

In the following, the LV distribution grid is defined as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the set of nodes (buses) is $\mathcal{V} = \{0, 1, \dots, n\}$ with

¹Code available on GitHub: <https://github.com/KIT-IAI/eGridLVGym>.

Table 1. An overview of closely related open source environments for flexibility in sustainable energy systems, highlighting key research gaps for RL in various real-world scenarios.

Related Works	Research Scope	Power Flow	Photo-voltaics (PV)	Battery Energy Storage (BES)	Battery Electric Vehicles (BEV)	Heat Pump (HP)	Parallel Execution	Implicit Flexibility	Explicit Flexibility	Supply-side Flexibility	Demand-side Flexibility
Vazquez-Canteli et al. (2020)	Building Energy Management	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Donnot (2020)	Transmission Grid Operation	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Pigott et al. (2021)	Voltage Regulation	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Henry et al. (2021)	Micro Grid Control	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Blum et al. (2021)	Building Energy Management	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Biagioni et al. (2022)	Multiagent for Any Energy Task	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Cui et al. (2022)	Load Frequency Control	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Zhang et al. (2023)	HVAC Control	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Yeh et al. (2023)	grid-to-vehicle (G2V) Control	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Orfanoudakis et al. (2024)	vehicle-to-grid (V2G) Control	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
The present paper	MORL Control	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

$n \in \mathbb{N}^+$. The undirected edges (or lines) connecting these nodes are given by the set of pairs such that $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. In this paper, we model the LV grid as a radial tree graph of $n + 1$ buses. The bus indexed as 0, representing the tree's root (the primary slack or substation bus connected to the superior grid level), regulates active and reactive power within the LV grid while maintaining a fixed reference voltage v_0 . We define the complete bus node set as the union of the subsets corresponding to each distribution grid element, $\mathcal{V} = \mathcal{V}_{\text{load}} \cup \mathcal{V}_{\text{HP}} \cup \mathcal{V}_{\text{PV}} \cup \mathcal{V}_{\text{BES}} \cup \mathcal{V}_{\text{BEV}}$, ensuring that each node in $\mathcal{V}$ belongs to at least one subset. Let $s_i = p_i + jq_i$ denote the net complex power injection at bus $i \in \mathcal{V}$. The voltage magnitude and phase angle at bus i are v_i and θ_i , respectively. Moreover, let $\theta_{ij} = \theta_i - \theta_j$ denote the phase angle difference between buses i and j . The parameters b_{ij} and g_{ij} represent the susceptance and conductance on the line $(i, j) \in \mathcal{E}$. By separating the active p_i and reactive q_i components, the power flow equations at bus i can be written as follows:

$$q_i = -v_i^2 \sum_{\mathcal{E}_{\pi(j)}} b_{ij} + v_i \sum_{\mathcal{E}_{\pi(j)}} v_j (g_{ij} \sin \theta_{ij} + b_{ij} \cos \theta_{ij}) \forall i \in \mathcal{V}, \quad (1a)$$

$$p_i = v_i^2 \sum_{\mathcal{E}_{\pi(j)}} g_{ij} - v_i \sum_{\mathcal{E}_{\pi(j)}} v_j (g_{ij} \cos \theta_{ij} + b_{ij} \sin \theta_{ij}) \forall i \in \mathcal{V}, \quad (1b)$$

where $\mathcal{E}_{\pi(j)} \subseteq \mathcal{V}$ is the set of nodes that are parents and/or children of bus j . Eqs. (1a) and (1b) represents the dynamics of the electrical grid, forming the basis for power flow and voltage control problems. These equations ensure grid stability and voltage limits between 0.9 and 1.1 per unit (pu), as defined in EN 50160 and VDE-AR-N 4105 [5, 14]. According to IEEE C57.91 and IEC 60076-7 [17, 19], cyclic transformer overloads of up to 150% and transient nature peak loads of up to 200% of rated capacity are permissible under normal operating conditions (see Appendix A.1). Flexibility in supply and demand helps ensure stable energy system operation. It includes measures such as PV curtailment, HP load shifting, BES use, and bi-directional BEV charging with explicit state of charge (SoC) targets, all while balancing user comfort and grid stability. Such combined multi-use energy control applications can be modeled by an optimal power flow formulation. Specifically, the objective function $J(\cdot)$ can be mathematically formulated as a linear program (LP) or quadratic program (QP) cost function, depending on the controllable decision variables $\mathbf{u}(k)$ (e.g., BES, HP setpoints) and the

uncontrollable state variables $\mathbf{x}(k)$ (e.g., load predictions, renewable generation). Predictions $\hat{\mathbf{z}}(k)$ summarize all future information. Let $\mathcal{H} = \{1, \dots, H_c\}$ represent all discrete time steps. The optimization problem is solved in a rolling horizon manner at each time step k within the finite control horizon H_c and can be formulated sequentially as:

$$\min_{\mathbf{u}} \sum_{k=1}^{H_c} J(\mathbf{x}(\mathbf{u}(k)), \mathbf{u}(k), \hat{\mathbf{z}}(k)) \quad (2a)$$

$$\text{s.t. } g_i(\mathbf{x}(\mathbf{u}(k))) \geq 0, \quad \forall i \in \mathcal{I}, \forall k \in \mathcal{H}, \quad (2b)$$

$$h_e(\mathbf{x}(\mathbf{u}(k))) = 0, \quad \forall e \in \mathcal{E}, \forall k \in \mathcal{H}, \quad (2c)$$

$$u_{\min}(k) \leq \mathbf{u}(k) \leq u_{\max}(k), \quad \forall k \in \mathcal{H}. \quad (2d)$$

The implicit function $\mathbf{x}(\mathbf{u}(k))$ describes the relationship between the state and the control variables. It is solved once during each iteration step of the optimization algorithm. Additionally, the initial control variable $\mathbf{u}_{\text{init}}(k)$ is given and used to solve the implicit function. The task of dispatchable power is given as an optimization problem to find the optimal setpoints for the heterogeneous DERs:

$$\mathbf{u}^* = \arg \min_{\mathbf{u}} J, \quad (3)$$

that minimizes a control objective function. Controller actions are iteratively adjusted in a sequential process, evaluated at fixed intervals. We formulate this problem as a POMDP [40] and solve it using SORL and MORL approaches (see Section 4).

3.2 Power Grids in eGridLVGym

As shown in Figure 2, the eGridLVGym framework for single and multi-objective versions with identical configurations but differing in their reward structures.

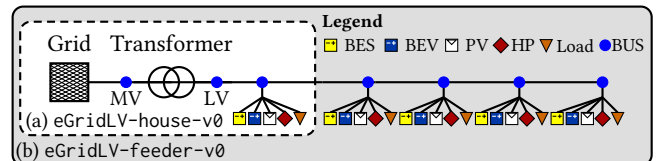


Fig. 2. Visualization of the eGridLVGym environments with single- and multi-objective reward versions.

3.2.1 Dispatchable Household (eGridLV-house-v0). Our first environment topology, shown within the dashed white box in Figure 2a, models a single household connected to a LV grid and the external medium-voltage grid via a transformer. It consists of a static load, a PV system, a BES, a BEV, and a HP. This topology represents a typical application of a home energy management system (HEMS) for minimizing household costs and/or emissions by maximizing the utilization of RES through aligning consumption with availability. Minor power fluctuations make voltage regulation negligible in this environment. This baseline scenario illustrates use cases such as shifting DER demand (HP, BEV) and maximizing self-consumption. It also demonstrates the flexibility of feeding energy back into the grid, guided by dynamic tariffs.

3.2.2 Dispatchable Feeder (eGridLV-feeder-v0). The second environment, enclosed within the gray box in Figure 2b, represents a single-feeder distribution grid. This topology features five points of Point of Common Coupling (PCC), illustrating the environment's broader connectivity and interaction points. Each PCC represents a household (or small group of households) with a static load, BEV, BES, HP, and PV system. Compared to the single household case, different loads and sizings increase the complexity of voltage regulation and coordinated dispatch.

3.3 Distributed Energy Resources Modeling

In this section we detail the modeling of various DER components (BEV, BES, HP, PV) and the inverter modeling for a partially observable energy system environment. Sensor modeling is described in Appendix A.3. At $k = 0$, the initial states of all DER components and associated systems are predefined as initial conditions.

3.3.1 Battery Electric Vehicle Modeling. The binary indicator variable $c_{\text{off},i}^{\text{BEV}}(k) \in \{0, 1\}$ denotes whether the BEV is available at time step k . When $c_{\text{off},i}^{\text{BEV}}(k) = 0$, the BEV is connected to the grid, and its SoC $E_i^{\text{BEV}}(k)$ is updated using the control variable $u_i^{\text{BEV}}(k) = p_i^{\text{BEV}}(k)$, representing charging or discharging power. When the BEV is disconnected ($c_{\text{off},i}^{\text{BEV}}(k) = 1$), charging and discharging are set to zero by the constraints. In the following, we describe for $i \in \mathcal{V}_{\text{BEV}}$ the BEV state, in- and equality constraints [1, 8]:

$$E_i^{\text{BEV}}(k+1) = E_i^{\text{BEV}}(k) - \eta_{\text{self},i}^{\text{BEV}} E_{\text{max},i}^{\text{BEV}} + p_i^{\text{BEV}}(k+1) (1 - c_{\text{off},i}^{\text{BEV}}(k+1)) \Delta t \quad \forall k \in \mathcal{H}, \quad (4a)$$

$$p_i^{\text{BEV}}(k) = \eta_{\text{cha}}^{\text{BEV}} p_{\text{cha}}^{\text{BEV}}(k+1) + \eta_{\text{dis}}^{\text{BEV}} p_{\text{dis}}^{\text{BEV}}(k+1) \quad \forall k \in \mathcal{H}, \quad (4b)$$

$$0 \leq -p_{\text{cha}}^{\text{BEV}}(k+1) \leq p_{\text{cha-max}}^{\text{BEV}} (1 - c_{\text{off},i}^{\text{BEV}}(k+1)) \quad \forall k \in \mathcal{H}, \quad (4c)$$

$$0 \leq p_{\text{dis}}^{\text{BEV}}(k+1) \leq p_{\text{dis-max}}^{\text{BEV}} (1 - c_{\text{off},i}^{\text{BEV}}(k+1)) \quad \forall k \in \mathcal{H}, \quad (4d)$$

$$E_{\text{min},i}^{\text{BEV}} \leq E_i^{\text{BEV}}(k+1) \leq E_{\text{max},i}^{\text{BEV}} \quad \forall k \in \mathcal{H}, \quad (4e)$$

$$p_{\text{cha}}^{\text{BEV}}(k+1) p_{\text{dis}}^{\text{BEV}}(k+1) = 0 \quad \forall k \in \mathcal{H}. \quad (4f)$$

3.3.2 Battery Energy Storage Modeling. The BES equations use the same mathematical formulations (4a-4f) to represent the behavior described in the BEV modeling from Section 3.3.1. The binary variable is set to a constant value of zero because the BES system is permanently connected.

3.3.3 Heat Pump Modeling. The power consumption denoted as $p_i^{\text{HP}}(k)$ of the HP can be modulated using the control variable $u_i^{\text{HP}}(k)$. For simplicity, the available control is not a free control of the power but instead a scaling of the originally scheduled time series. This enables load shifting but retains the original usage patterns and total power consumption. The HP control has to satisfy the following in- and equality constraints for $i \in \mathcal{V}_{\text{HP}}$:

$$p_i^{\text{HP}}(k) = u_i^{\text{HP}}(k) \cdot p_{\text{orig},i}^{\text{HP}}(k) \quad \forall k \in \mathcal{H}, \quad (5a)$$

$$0.5 \leq u_i^{\text{HP}}(k) \leq 1.5 \quad \forall k \in \mathcal{H}, \quad (5b)$$

$$0 \leq p_i^{\text{HP}}(k) \leq p_{\text{max},i}^{\text{HP}} \quad \forall k \in \mathcal{H}, \quad (5c)$$

$$\Delta p_{\text{heat}} = \sum_{j=1}^{N_{24h}} (p_{\text{orig},i}^{\text{HP}}(k+1-j) - p_i^{\text{HP}}(k+1-j)) \quad \forall k \in \mathcal{H}. \quad (5d)$$

Eq. (5a) describes the power consumption of the HP at time k . When the HP is in operation, the power consumption $p_i^{\text{HP}}(k)$ is equal to the control variable $u_i^{\text{HP}}(k)$ multiplied by the original rated power $p_{\text{orig},i}^{\text{HP}}(k)$, which is taken from the load prediction. Eq. (5b) restricts control actions to prevent disabling or over-scaling the HP. Eq. (5c) ensures the net power remains non-negative and within the HP's maximum rating. Finally, Eq. (5d) tracks the net heat consumption deviation over the last 24 hours.

3.3.4 Inverter Modeling. If reactive power is part of the control scheme, it can be managed by supporting components (BES and PV inverters). The reactive power demand is set by adjusting the power factor $\cos \varphi$ in Eq. (6a). According to the grid code [14], each PV or BES unit has a minimum power factor $\min \cos \varphi$ in overexcited or underexcited conditions. This value is set to 0.90 or 0.95, depending on the maximum apparent power as defined in Eq. (6e). If the reactive power demand plus the active power exceeds the available apparent power, the active power is curtailed continuously in line with Eq. (6c). Grid codes restrict reactive power to 10% of active power in Eq. (6f), as long as active power is below 10% (or 20%, depending on generator size) of the rated power.

$$P_i(k) = S_i(k) * \cos \varphi \quad \forall k \in \mathcal{H}, \quad (6a)$$

$$Q_i(k) = S_i(k) * \sin \varphi \quad \forall k \in \mathcal{H}, \quad (6b)$$

$$S_i(k) \leq S_{i,\text{max}} \quad \forall k \in \mathcal{H}, \quad (6c)$$

$$S_i(k) = \sqrt{P_i(k)^2 + Q_i(k)^2} \quad \forall k \in \mathcal{H}, \quad (6d)$$

$$\cos \varphi_i(k) \geq \min \cos \varphi_i \quad \forall k \in \mathcal{H}, \quad (6e)$$

$$P_i(k) * 0.1 \geq Q_i(k) \text{ if } P_i(k) \leq 0.2 * P_{\text{max}} \quad \forall k \in \mathcal{H}. \quad (6f)$$

4 Reinforcement Learning Control Settings

The increasing automation of energy system dynamics, driven by the ongoing energy transition in combination with the faster integration of RES coupled with higher degrees of freedom in the controller, makes it more difficult to solve the optimization problem in Eqs. (2a-2d) using conventional methods (e.g. nonlinear programming). Instead, we frame this optimal control as a sequential decision-making problem [40], modeling the grid planning for DERs in a distribution grid as a POMDP [4, 38, 39]. A POMDP can be mathematically described by a 7-tuple $(\mathcal{S}, \mathcal{A}, \mathcal{T}, R, O, \Omega, \gamma)$. We consider the state space $\mathcal{S}$ and the action space $\mathcal{A}$ to be continuous. The state

Table 2. Reinforcement learning control notations.

Symbol	Description
$s_k \in \mathcal{S}$	System state at timestep k
$o_k \in \mathcal{O}$	Observation (partial) at timestep k
$a_k \in \mathcal{A}$	Action selected at timestep k
$\pi(a o)$	Control policy (mapping observations to actions)
H_c, H_p	Control and prediction horizons (episode, forecast)
$\mathbf{a}_k$	Action vector in $[-1, 1]$ (active/reactive setpoints)
$\mathbf{p}_{\text{BES/BEV/HP}}^{\text{BES/BEV/HP}}$	Active power vector for the BES, BEV, and HP
$\mathbf{q}_{\text{BES/PV}}^{\text{BES/PV}}$	Reactive power vector for the BES and PV inverters
r_k^i	Reward component i at timestep k
Δ_k^ℓ	Penalty term ℓ at timestep k
λ_i, μ_ℓ	Weighting factors for rewards and penalties
$\text{SoC}_k^{\text{BEV}}$	State of charge of the BEV battery at timestep k
v_k, v_{ref}	Actual and reference voltage at timestep k
$E_k^{\text{BES}}, E_{\text{max}}^{\text{BES}}$	Current/Maximum battery energy
$\Delta p_{\text{heat},k}$	Net heat consumption deviation from originally

transition function $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$ defines the probability density of transitioning from state $s_k \in \mathcal{S}$ to state $s_{k+1} \in \mathcal{S}$ given an action $a_k \in \mathcal{A}$. With each transition, the environment returns an observation $o_k \in \mathcal{O}$ and a reward based on the global reward function $R : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$, which evaluates the RL agent's decision. The observation probability function $\Omega : \mathcal{S} \times \mathcal{A} \times \mathcal{O} \rightarrow [0, 1]$ captures the likelihood of observing o_{k+1} given state s_{k+1} and action a_k . The discount factor $\gamma \in [0, 1)$ determines the weighting of future rewards. The agent receives o_k for partially observable settings, which does not fully describe the state s_k . The goal of a RL agent is to learn an optimal control policy π^* that maximizes the expected return over an episode with a finite control horizon H_c :

$$\pi^*(a|o) = \arg \max_{\pi} \mathbb{E}_{\pi} \left[\sum_{k=0}^{H_c} \gamma^k R_k \right]. \quad (7)$$

At every timestep k , the agent observes $\mathbf{o}_k$, selects $\mathbf{a}_k$ from the action space, and obtains a reward R_k . This process is repeated until a terminal condition is met or the episode is terminated (e.g. if the power flow calculation does not converge). We set the prediction horizon H_p to eight time steps (two hours). By default, one episode spans seven days with a 15-minute resolution, resulting in 672 steps. An overview of the RL control notation is given in Table 2.

4.1 State and Observation Space

The states $\mathcal{S}$ and observations $\mathcal{O}$ are continuous and normalized to $[-1, 1]$ using min-max scaling. The system state consists of baseline and optionally extended observations. In Table 4, predicted values are exclusively used for the agent's observation of future PV, load, and energy prices, while reward computations use actual values at each time step k .

Baseline observations (#1–9 in Table 3). These include the predicted load, HP and PV power profiles over the prediction horizon H_p ; the current storage states for HPs, BESs, and BEVs; and temporal

indicators for the time of day, day of the year, and binary flags for BEV availability. Hence, the dimension of the baseline observation is determined by the number of DERs $\mathcal{D}$ plus the two temporal indicators (time and date).

Extended observations. Extended observations $\mathbf{s}_k^{\text{ext}}$ can be selectively appended to the baseline observation vector $\mathbf{s}_k^{\text{base}}$ to form the complete observation $\mathbf{s}_k$, as shown in Eq. (8).

$$\mathbf{s}_k = (\mathbf{s}_k^{\text{base}}, \mathbf{s}_k^{\text{ext}}) = \begin{cases} (\mathbf{s}_k^{\text{base}}, \mathbf{s}_k^{\text{economic}}), & (\text{economic: \#1–10}) \\ (\mathbf{s}_k^{\text{base}}, \mathbf{s}_k^{\text{voltage}}), & (\text{voltage: \#1–9 + \#11}) \\ (\mathbf{s}_k^{\text{base}}, \mathbf{s}_k^{\text{economic}}, \mathbf{s}_k^{\text{voltage}}), & (\text{combined: all\#}) \end{cases} \quad (8)$$

Sensor noise (see Section A.3) is added to mimic real-world measurements. The agent does not receive the step index within an episode or any past states, preventing it from exploiting episode boundaries (e.g., fully discharging a battery at the end). Depending on the application, the extended observations can include economic, voltage, or combined information. This enables baseline, extended (economic or voltage), or combined observation modes. Table 3 lists all variables (#1–9: baseline, #10: economic, #11: voltage, and all#: combined), including their dimensions, types, ranges, and noise settings. All observations can be concatenated into a single vector for compatibility with RL libraries or stored in a dictionary structure.

4.2 Action Space

At each time step k , the RL agent operates within the continuous joint action set of DERs is given as $\mathbf{a}_k \in \mathcal{A} = \times_{i \in \mathcal{D}} \mathcal{A}_i$, where $\mathcal{D}$ represents the total DER units in the environment. The dimension and structure of the agent's action $\mathbf{a}_k$ depend on the specific use case. Accordingly, $\mathbf{a}_k$ is defined as follows:

$$\mathbf{a}_k = \begin{cases} (\mathbf{p}_{\text{BES}}^{\text{BES}}, \mathbf{p}_{\text{BEV}}^{\text{BEV}}, \mathbf{p}_{\text{PV}}^{\text{PV}}) \in \Theta^{3|\mathcal{V}|}, & (\text{economic/voltage}) \\ (\mathbf{p}_{\text{BES}}^{\text{BES}}, \mathbf{p}_{\text{BEV}}^{\text{BEV}}, \mathbf{p}_{\text{HP}}^{\text{HP}}, \mathbf{q}_{\text{BES}}^{\text{BES}}(\mathbf{v}), \mathbf{q}_{\text{PV}}^{\text{PV}}(\mathbf{v})) \in \Theta^{5|\mathcal{V}|}, & (\text{shielding}) \\ (\mathbf{p}_{\text{BES}}^{\text{BES}}, \mathbf{p}_{\text{BEV}}^{\text{BEV}}, \mathbf{p}_{\text{HP}}^{\text{HP}}, \mathbf{q}_{\text{BES}}^{\text{BES}}, \mathbf{q}_{\text{PV}}^{\text{PV}}) \in \Theta^{5|\mathcal{V}|}, & (\text{combined}) \end{cases} \quad (9)$$

Each action component is normalized to $\Theta := [-1, 1]$ and mapped to physically valid setpoints (e.g., real or reactive power limits) to ensure feasibility. In the economic use case, for instance, $p_{\text{BES}}^{\text{BES}} > 0$ indicates that the battery is charging, whereas $p_{\text{BES}}^{\text{BES}} < 0$ indicates discharging. Similarly, heat pumps use the range Θ to scale their load factor within $[0.5, 1.5]$, allowing a portion of the load to be shifted over time (e.g., rescheduling a heating cycle) while distorting the main load profile. If a physical constraint is violated, the controller adjusts the action again and returns a penalty.

4.3 Reward Function

The total scalar reward at each timestep k is

$$R_k = \sum_{i=1}^I \lambda_i r_k^i + \sum_{\ell=1}^L \mu_\ell \Delta_k^\ell, \quad r_k^i, \Delta_k^\ell \in \Theta. \quad (10)$$

Here, r_k^i are reward components (weighted by λ_i), and Δ_k^ℓ are penalty terms (weighted by μ_ℓ). Depending on the selected RL control mode in Table 4, different subsets of $\{r_k^i\}$ are activated. Reward components are discussed in detail in Appendix A.5.

Table 3. Observation spaces for the baseline (#1–9), extended economic (#1–10), voltage (#1–9 and #11), and combined (all #).

#	Variable	Description	Dimension	Type	Range	Noise Added
1	pv_power	Predicted PV power values.	$(H_p, \mathcal{V})$	Continuous	$[0, 1]$	Yes
2	load_power	Predicted load power profiles.	$(H_p, \mathcal{V})$	Continuous	$[0, 1]$	Yes
3	heating_power	Predicted heating power values.	$(H_p, \mathcal{V})$	Continuous	$[0, 1]$	Yes
4	heat_storage	Thermal storage state.	$(1, \mathcal{V})$	Continuous	$[0, 1]$	No
5	soc_percent	State of charge for BES.	$(1, \mathcal{V})$	Continuous	$[0, 1]$	No
6	bev_soc_percent	State of charge for BEV.	$(1, \mathcal{V})$	Continuous	$[0, 1]$	No
7	bev_present	Binary indicator for BEV presence.	$(1, \mathcal{V})$	Binary	$\{0, 1\}$	No
8	time_of_day	Normalized time of day.	(1)	Continuous	$[0, 1]$	No
9	date	Normalized day index.	(1)	Continuous	$[0, 1]$	No
10	price	Predicted electricity prices over the horizon.	$(H_p, 1)$	Continuous	$[0, 1]$	Yes
11	voltages	Normalized voltage magnitudes.	$(1, \mathcal{V})$	Continuous	$[0, 1]$	Yes
12	line_loading	Normalized line loading.	$(1, \mathcal{E})$	Continuous	$[0, 1]$	No
13	transformer_loading	Normalized transformer loading.	(1)	Continuous	$[0, 1]$	No

Reward Components.

$$r_k^{\text{economic}} = -p_k^{\text{grid}} \text{price}_k, \quad (11)$$

$$r_k^{\text{flexible}} = \begin{cases} \text{SoC}_k^{\text{BEV}} - 0.5, & \text{if BEV present,} \\ 0, & \text{otherwise,} \end{cases} \quad (12)$$

$$r_k^{\text{voltage}} = - \sum_{i \in \mathcal{V}_{\text{BUS}}} (v_{i,k} - v_0)^2, \quad (13)$$

$$r_k^{\text{curtailment}} = - \sum_{i \in \mathcal{V}_{\text{PV}}} (p_{i,k}^{\text{PV}} - p_{i,k}^{\text{curtailment}}), \quad (14)$$

$$r_k^{\text{loading}} = - \left(\ell_k^{\text{transformer}} + \sum_{j \in \mathcal{E}} \ell_{j,k}^{\text{line}} \right). \quad (15)$$

Soft Physical Penalties.

$$\Delta_k^{\text{BES}} = \begin{cases} -1, & \text{if } E_k^{\text{BES}} > E_{\text{max}}^{\text{BES}} \text{ or } E_k^{\text{BES}} < 0, \\ 0, & \text{otherwise,} \end{cases} \quad (16)$$

$$\Delta_k^{\text{HP}} = \begin{cases} |\Delta p_{\text{heat},k}| - p_{\text{band}}, & \text{if } \Delta p_{\text{heat},k} > p_{\text{band}}, \\ 0, & \text{otherwise.} \end{cases} \quad (17)$$

4.4 Reinforcement Learning Algorithms

This section first presents eight SORL methods as state-of-the-art examples for different algorithmic architectures, followed by a MORL approach. For completeness, Appendix A.2 describes a rule-based control (RBC) reference controller, and Appendix A.4 details the hyperparameters for both SORL and MORL algorithms.

4.4.1 Single-Objective RL Algorithms.

- **DDPG and TD3 (off-policy actor-critic):** Deep Deterministic Policy Gradient (DDPG) learns a deterministic policy directly, while Twin Delayed DDPG (TD3) employs two independent Q-networks and delayed target updates to reduce Q-value overestimation and stabilize training.
- **SAC and TQC (off-policy):** Soft Actor-Critic (SAC) maximizes policy entropy, balancing exploration and exploitation. Truncated Quantile Critics (TQC) extends SAC by using multiple critics based on Q-Learning and quantile regression to model the return distribution. TQC truncates high Q-values (overestimations) to lower variance and training stability.

- **A2C and TRPO (on-policy):** Advantage Actor-Critic (A2C) updates actor (policy) and critic (value function) simultaneously, leveraging the advantage function to reduce variance in the policy gradient and improve training stability. Trust Region Policy Optimization (TRPO) constrains policy updates to a trust region, to avoid large performance drops but higher computational cost.
- **PPO and Recurrent PPO (on-policy):** Proximal Policy Optimization (PPO) simplifies TRPO’s trust region mechanism with a clipped objective, providing stable efficient updates. Recurrent PPO integrates recurrent network architectures (e.g., LSTM) to handle partial observability and extend temporal dependencies.

4.4.2 Multi-Objective RL Algorithm. Instead of summing multiple reward components into a single scalar, MORL maintains them as a vector. Each objective (e.g., economic cost, voltage stability, flexibility) is a separate dimension in the reward vector:

$$\mathbf{R}_k = \left(r_k^{\text{economic}} \quad r_k^{\text{flexible}} \quad \sum_{\ell=1}^L \mu_{\ell} \Delta_k^{\ell} \right)^{\top}. \quad (18)$$

By preserving individual objectives, MORL can learn to achieve Pareto-optimal solutions and later accommodate changing trade-offs. In contrast, SORL must fix and sum these objectives beforehand, losing the flexibility to adjust weights post-training.

Concave-Augmented Pareto Q-Learning (CAPQL). CAPQL extends Q-learning to handle unknown or varying preference weights $w \in \Phi$ [25]. The vectorial reward function $\mathbf{R}(s, a)$, mapping $\mathbf{R} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}^d$ with d objectives, is augmented by a strongly concave term (e.g., entropy) to smooth updates and reduce instability. Both the Q-network and policy are conditioned on w that defines the scalarization of the multi-objective reward. During training, $w \in \Phi$ is sampled at each step, allowing the networks to generalize across different trade-offs between objectives and allowing the policy to approximate the *Pareto front*. Concave regularization keeps the target value function continuous in w , improving convergence speed and numerical stability. For a given w and discount factor γ , the CAPQL objective is:

$$\pi^*(\cdot; w) = \arg \max_{\pi(\cdot; w)} \mathbb{E} \left[\sum_{k=0}^{\infty} \gamma^k (w^{\top} \mathbf{R}(s_k, a_k) + \alpha H(\pi(s_k; w))) \right] \quad (19)$$

Table 4. Overview of RL control modes.

#	Mode	Action (Goal) (Variable)	Inverter Constraint	Observation and Reward* Terms
1	economic	p	No	#1–10; Eqs. (11–12)
2	voltage	p	No	#1–9, #11; Eqs. (13–14)
3	shielding	$p, q(v)$	explicit (hard)	#1–11; Eqs. (11–12)
4	combined	p, q	implicit (soft)	all #; Eqs. (11–15)

* includes soft physical penalties for BES and HP, see Eqs. (16–17).

where the hyperparameter $\alpha > 0$ is a weighting coefficient for the concave entropy term $H(\pi)$. By learning $Q(s, a, w)$ and $\pi(s; w)$ jointly for all w , CAPQL can adapt to multiple objectives under dynamic priorities. With this definition SORL is a special case of MORL (fixed weights), whereas CAPQL explores the entire Pareto front, offering greater flexibility and robustness.

4.5 Reinforcement Learning Control Modes

We define four RL control mode objectives for heterogeneous DERs: economic, voltage, shielding, and combined. Table 4 summarizes their actions, inverter constraints, and observation/reward terms. Violations of grid code constraints by any mode are overridden by the hard constraint controller (see Appendix A.1).

Economic Mode. In economic mode, the RL agent controls only active power, with reactive power set to zero, aiming to maximize economic rewards defined by Eqs. (11–12). The observation space comprises baseline variables (#1–9) and price-related features (#10).

Voltage Mode. In voltage mode, the agent similarly controls only active power. In this mode, voltage measurements (#11) replace price-related features within the observation space. The reward function explicitly targets minimization of voltage deviations and PV curtailment penalties, as described in Eqs. (13–14).

Shielding Mode. In shielding mode, the RL agent controls the active power, while the reactive power is controlled over an explicit shielding mechanism according to a configurable piecewise linear function $q(v)$. We implement VDE-AR-N 4105 [14] in response to the real-time voltage measurements. This shielding mechanism on PCC level imposes hard constraints to prevent voltage violations. It increases the apparent power and power flow exchange in the grid, potentially leading to a higher transformer load.

Combined Mode. In combined mode, the RL agent jointly regulates active and reactive power, providing maximal flexibility for balancing economic objectives and complying with grid constraints (e.g., voltage stability and transformer loading). In contrast to shielding mode, combined mode employs soft constraints for reactive power control on inverters, as detailed in Section 3.3.4. These soft constraints comply with inverter regulations specified by VDE-AR-N 4105 [14], ensuring reactive power stays within regulatory limits and can be adapted to similar international standards (e.g., IEEE 1547-2018 [18]). Compared to hard constraints in shielded mode, this approach offers greater flexibility, enabling smarter coordination across multiple inverter units and maximizing total rewards.

5 Evaluation

5.1 Experimental Settings

We use the energy system environments described in Section 3.2. These simulation environments include the loads and PV profiles from [27]. To model variable and realistic charging station demands, we use daily commuter mobility data from [41], supplemented by Ile-de-France survey data, and we extrapolate hourly event probabilities to quarter-hour intervals. Each PCC has a BEV wallbox that operated with a maximum output of 8 kW and able to charge one car at a time. At each time step, a departure event occurs if a car is present at the charger, or an arrival event occurs otherwise. Each charging point has its own independent arrival and departure events. Based on this model, each BEV takes about 0.8 trips per day. A recent analysis of the registration data in [11] and [20] for 2025 indicated that BEV with battery capacities between 60 and 80 kWh currently dominate the market. Therefore, we set a capacity of 75 kWh per vehicle. We use German electricity price data from the EPEX SPOT platform for 2023 to inform the hourly price scheme [29]. We align the hourly prices with the 15-minute load and PV profiles, and trim the dataset at the 15th and 85th percentiles to remove outliers (negative prices remain). We explicitly performed this preprocessing step to stabilize the learning dynamics and prevent policy distortions caused by rare extreme price spikes. We evaluate the performance of state-of-the-art SORL algorithms, including DDPG [24], TD3 [13], SAC [15], TQC [22], A2C [28], TRPO [36], and PPO [35], as well as the SORL algorithms with grid code shielding. In the MORL setting, we evaluate the algorithm CAPQL [25] on this dynamic energy distribution problem with continuous DER actions. In our

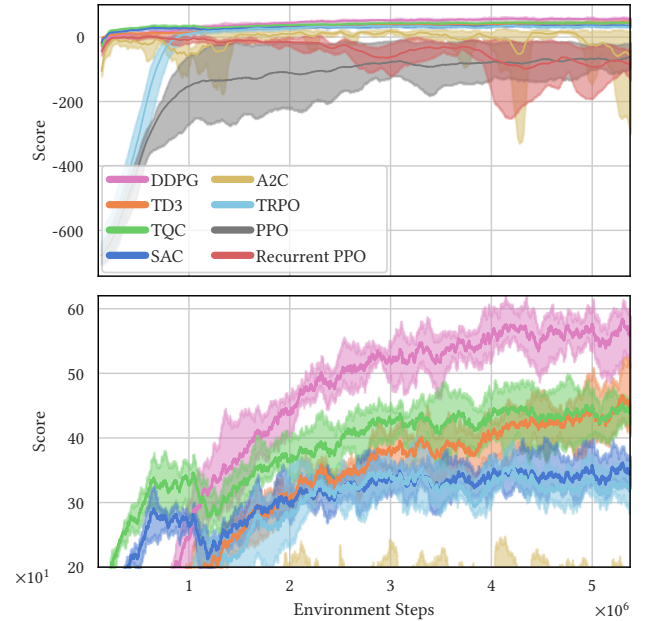


Fig. 3. Training curves for continuous DER control in the eGridLV-house-v0 environment. Solid lines show the mean across 5 runs with different seeds and fixed hyperparameters.

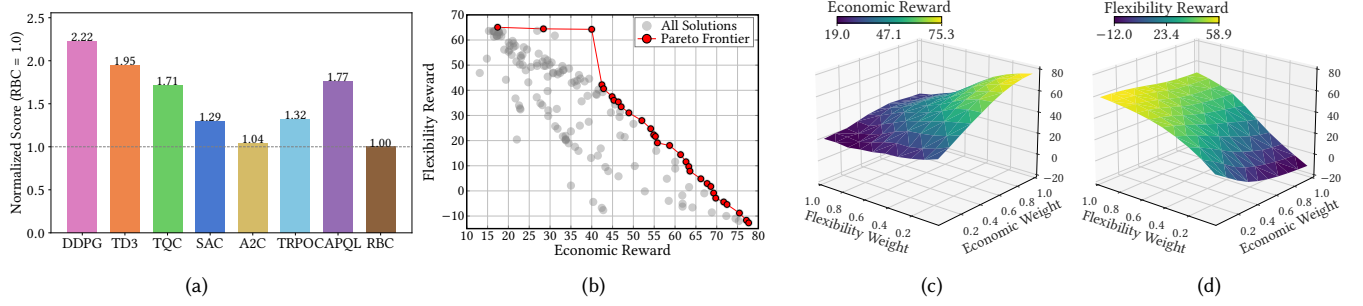


Fig. 4. CAPQL performance in the eGridLV-house-v0 environment under economic mode. Subfigures: (a) comparison with SORL algorithms, (b) Pareto frontier analysis, (c) economic, and (d) flexibility reward surface across weight settings.

experiments, we explore different RL control modes using the rewards or metrics shown in Table 4. We split the 51-week dataset into 40 weeks for training and 11 weeks for evaluation, using the same seed across all algorithms for both training and evaluation. Each training episode is randomly sampled from the training set based on this seed. We run 8000 training episodes in total, corresponding to $8000 \times 672 = 5,376,000$ environment steps. During training, we test the evaluation set once every 500 episodes to monitor global progress and detect overfitting. No back-propagation is performed on the evaluation set to avoid inadvertently training on it. We train ten algorithm variants in total, largely relying on default hyperparameters from the SB3 repository. For each algorithm, we run five different random seeds. Whenever possible, we employ state-dependent exploration, with slight rounding of batch sizes and learning starts at either 96 or 672 steps.

5.2 Performance Comparison on Control Modes

This section demonstrates the eGridLVGym framework and its integration with SORL and MORL algorithms, while possible future research directions such as XRL can be investigated.

5.2.1 Economic Mode. We first present the training results of all algorithms in the economic mode, as shown in Figure 3. Four SORL algorithms (DDPG, TD3, SAC, and TQC) achieve competitive and stable solutions after two million steps, despite some slight instability. In contrast, PPO fails to solve the environment with the standard parameters, while Recurrent PPO, A2C, and TRPO solve the environment but yield significantly lower rewards. To evaluate CAPQL’s performance, we set all weights to one, as in the original SORL setting. As illustrated in Figure 4a, CAPQL and other SORL algorithms converge to stable solutions, while default PPO and Recurrent PPO yield net negative rewards and are therefore excluded. Compared to the RBC controller, DDPG achieves a 122% performance increase, while CAPQL shows a 77% increase. This improvement is achieved despite the fact that RL algorithms, including CAPQL, incur penalties for soft physical violations. In contrast, the RBC avoids these penalties but is less effective overall. The main advantage of CAPQL and other MORL algorithms is the ability to adapt the weighting of multiple objective functions. To construct the Pareto frontier, we evaluate 164 weighting combinations: 100 combinations are formed as cartesian product $[0.02, 0.13, 0.24, \dots, 1]^2$. The remaining

64 as $[0.01, 0.027, 0.193, \dots, 10]^2$. In doing so, both fine and broad trade-offs are covered. Figure 4b illustrates the resulting weight distribution in the objective space, with the Pareto frontier shown in red and all other solutions in grey. CAPQL reliably identifies efficient trade-offs ranging from low-cost, low-flexibility to high-flexibility, higher-cost solutions, demonstrating its ability to span a broad spectrum of preferences. Figures 4c–d depict the 3D reward surfaces based on linear grid weighting, highlighting CAPQL’s ability to adjust to varying priorities. This allows operators to shift any objective weights dynamically in response to external signals or changing grid conditions without retraining the agent.

5.2.2 Voltage Mode. Figure 5 compares the average voltage and PV curtailment performance of unscaled and scaled SORL algorithms against the CAPQL and a baseline reactive power controller. The evaluation is conducted in the eGridLV-feeder-v0 environment. Reward distributions are visualized with confidence intervals. While the grid code approaches require no PV curtailment, they exhibit higher voltage deviations. In contrast, all SORL agents improved voltage regulation, though at a minor increase in PV curtailment (below 7%). Among these, TQC achieves the lowest voltage deviation with a total PV curtailment of 6.19%, while SAC achieves 6.53%. To reduce curtailment, we increase the PV-curtailment weight ten-fold for the two most successful SORL algorithms. This adjustment reduces TQC curtailment to 5.12% and SAC to 5.74%, with a slight increase in voltage deviation that remains below the RBC. CAPQL achieves 0.76% curtailment with moderate voltage performance. In the MORL setting, CAPQL with adaptive weighting (voltage = 1, curtailment = 10) yields similar results. This adjustment requires no re-training, an advantage in real-world operations where priorities (e.g., stricter voltage limits) can shift dynamically.

5.2.3 Economic, Shielding and Combined Mode. In the most complex scenario, we jointly optimize economic objectives, ensure BEV charging, and manage active power setpoints for BES, BEVs, and HPs to maintain acceptable voltage levels and minimize PV curtailment. The large observation and action spaces, combined with a scalar reward that aggregates multiple objectives, make it challenging to trace individual DER control actions to specific outcomes. Figure 6 compares bus voltages (top) and transformer load (bottom) across

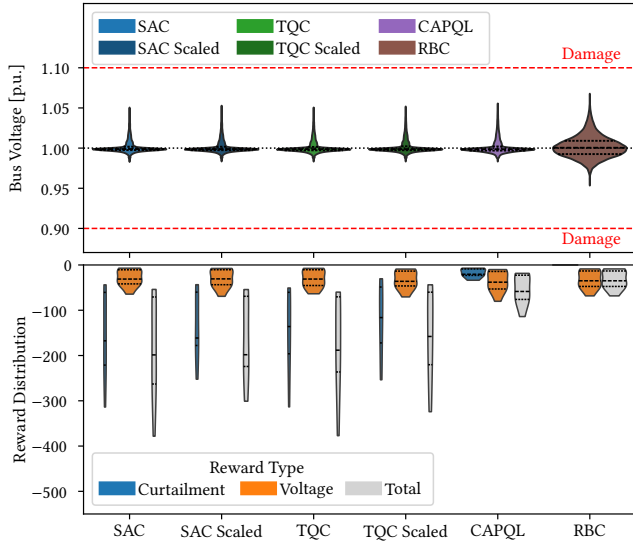


Fig. 5. Voltage mode comparison of SORL variants and MORL-based CAPQL in eGridLV-feeder-v0. Top: Bus voltage levels (EN 50160). Bottom: Reward distribution.

three RL modes: (i) an economic mode optimizing profit, (ii) a shielding mode optimizing profit subject to hard grid code voltage constraints, and (iii) a combined mode balancing economic and stability objectives over soft constraints. In this scenario, the IEC 60076-7 hard constraint controller for grid code compliance remains inactive, as the loading does not exceed the specified threshold. See Appendix A.1 for an example. Four algorithms (SAC, TD3, TQC, TRPO) show the most promising and similar learning performances. Even under the economic mode, A2C and DDPG, the bus voltages reached their highest point only in one timestep (1.054 p.u.), indicating that the grid remains robust under dynamic price signals. By contrast, the shielding mode consistently limits the minimum voltage for all algorithms and narrows the voltage band. Across most algorithms, the combined mode further narrows the voltage range—though at the cost of PV curtailment. Increasing the PV curtailment weight further reduces curtailment. As shown at the bottom of Figure 6, all combined modes significantly reduce transformer loading compared to economic and shielding modes. This is because combined mode leverages flexible reactive power control, while economic mode uses only active power, and shielding mode adds hard reactive power constraints. For instance, A2C reduces peak loading from 93% to 75% in combined mode. All algorithms maintain transformer loading below 100%, indicating effective control performance. Shielding mode stabilizes voltage with reactive power control while increasing transformer load caused by reactive power flow. Consequently, the voltage ranges in all algorithms are smaller than in economy mode, but the transformer load is higher. The amount of expected overloading can be adjusted by adapting the weighting factors of reward and/or penalty parameters in Eqs. (11–17). This illustrates the trade-offs in complex grid operations. While this improved grid stability slightly reduces profitability, the net benefit of curbing

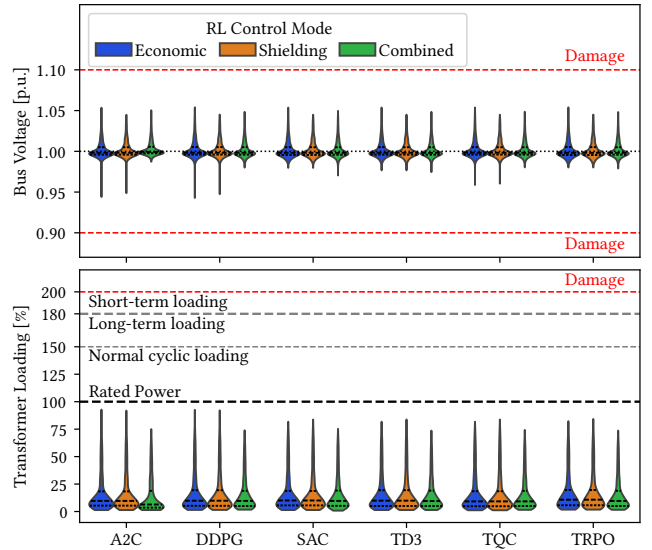


Fig. 6. RL modes performance in the eGridLV-feeder-v0 environment. Top: Bus voltage levels (EN 50160). Bottom: Transformer loading (IEC 60076-7).

costly overloads and voltage violations may outweigh the loss. In contrast to recent MARL studies on power grids, such as [2] and [45], our approach explicitly considers hard grid constraints with approved international grid codes, price signals, and the heterogeneity of DERs. Previous work has focused on homogeneous types of DERs (HP or PV), ignoring economic signals and neglecting grid constraints.

6 Conclusion

The present paper introduces eGridLVGym, a novel multi-objective RL environment for distributed energy systems (photovoltaics, battery storages, heat pumps, and electric vehicles). It offers single- and multi-objective RL variants, enabling the energy research community to tackle advanced tasks and explore innovative RL approaches.

Our empirical evaluations demonstrate that while state-of-the-art single-objective RL algorithms can handle scalarized multi-objective tasks, their adaptability to shifting priority weights is limited. In contrast, CAPQL overcomes this limitation by learning Pareto-optimal solutions and adapting trade-offs without retraining. RL shielding enforces voltage and device constraints through grid code guidelines to ensure operational safety while aiming to maximize revenues.

By addressing the present paper’s research questions, we demonstrate that RL can manage the often conflicting objectives of economic cost, voltage stability, and flexibility. We also show how various SORL algorithms perform under scalarized reward settings, confirm that grid code-based shielding reliably enforces operational safety, and establish MORL’s superior adaptability compared to SORL algorithms. eGridLVGym is a robust and versatile framework that enables the energy research community to explore, evaluate, and test new research questions and RL algorithms for the next generation of smart grids. Future research will focus on integrating

multi-dispatchable feeders, scaling to larger grids, dealing with forecast uncertainties, and the application of multi-objective multi-agent reinforcement learning (MOMARL) grid control.

Acknowledgments

This work was supported in part by the Energy System Design (ESD) Project; in part by the Helmholtz Association's Initiative and Networking Fund through Helmholtz AI and the HAICORE@KIT partition.

References

- [1] Anaïs Berkes and Srinivasan Keshav. 2024. SOPEVS: Sizing and Operation of PV-EV-Integrated Modern Homes. In *Proc. 15th ACM Int. Conf. on Future and Sustainable Energy Systems (e-Energy '24)* (Singapore, Singapore). ACM, New York, NY, USA, 14–26. <https://doi.org/10.1145/3632775.3661941>
- [2] David Biagioni et al. 2022. PowerGridworld: a framework for multi-agent reinforcement learning in power systems. In *Proc. 13th ACM Int. Conf. Future Energy Syst. (Virtual Event) (e-Energy '22)*. ACM, New York, NY, USA, 565–570.
- [3] David Blum et al. 2021. Building optimization testing framework (BOPTEST) for simulation-based benchmarking of control strategies in buildings. *J. Build. Perform. Simul.* 14, 5 (2021), 586–610.
- [4] Di Cao et al. 2020. Reinforcement Learning and Its Applications in Modern Power and Energy Systems: A Review. *Journal of Modern Power Systems and Clean Energy* 8, 6 (2020), 1029–1042. doi:10.35833/MPCE.2020.000552
- [5] CENELEC. 2022. *DIN EN 50160:2022-10, Voltage Characteristics of Electricity Supply by Public Distribution Networks*. Technical Report EN 50160:2022-10. Beuth Verlag GmbH, Berlin, Germany. <https://dx.doi.org/10.31030/3383427>
- [6] Xin Chen et al. 2022. Reinforcement Learning for Selective Key Applications in Power Systems: Recent Advances and Future Challenges. *IEEE Transactions on Smart Grid* 13, 4 (2022), 2935–2958. doi:10.1109/TSG.2022.3154718
- [7] Hantao Cui et al. 2022. Andes_gym: A Versatile Environment for Deep Reinforcement Learning in Power Systems. In *2022 IEEE Power & Energy Society General Meeting (PESGM)*. IEEE, Denver, CO, USA, 01–05.
- [8] Gökhan Demirel et al. 2025. PIDDE: Photovoltaic Integration Dynamics and Efficiency for Autonomous Control on Power Distribution Grids. *Energy Informatics* 8, 1 (March 2025), 32. doi:10.1186/s42162-025-00489-6
- [9] B. Donnot. 2020. Grid2op - A testbed platform to model sequential decision making in power systems. <https://GitHub.com/rte-france/grid2op>.
- [10] Energy and Inc. Environmental Economics. 2020. *Pathways to Deep Decarbonization in New York State*. Technical Report. NYSEERDA, San Francisco, CA.
- [11] Federal Motor Transport Authority (KBA). 2025. *Fahrzeuge - Neuzulassungen nach Umwelt-Merkmalen (FZ 28)*. https://www.kba.de/DE/Statistik/Produktkatalog/produkte/Fahrzeuge/fz28/fz28_gentab.html.
- [12] Florian Felten et al. 2024. A Toolkit for Reliable Benchmarking and Research in Multi-Objective Reinforcement Learning. In *Proc. 37th Int. Conf. on NeurIPS '23* (New Orleans, LA, USA). Curran Associates Inc., Red Hook, NY, USA, 30 pages.
- [13] Scott Fujimoto et al. 2018. Addressing Function Approximation Error in Actor-Critic Methods. arXiv:1802.09477 <https://arxiv.org/abs/1802.09477>
- [14] German Association for Electrical, Electronic and Information Technologies (VDE). 2018. *Generators Connected to the Low-Voltage Distribution Network*. VDE Association for Electrical, Electronic and Information Technologies, Frankfurt am Main. VDE-AR-N 4105:2018-11.
- [15] Tuomas Haarnoja et al. 2018. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. arXiv:1801.01290
- [16] Robin Henry et al. 2021. Gym-ANM: Reinforcement learning environments for active network management tasks in electricity distribution systems. *Energy and AI* 5 (Sept. 2021), 100092. doi:10.1016/j.egyai.2021.100092
- [17] IEEE. 2012. IEEE Std C57.91-2011: IEEE Guide for Loading Mineral-Oil-Immersed Transformers and Step-Voltage Regulators. IEEE Std, 123 pages.
- [18] IEEE. 2018. *IEEE Standard for Interconnection and Interoperability of Distributed Energy Resources with Associated Electric Power Systems Interfaces*. Institute of Electrical and Electronics Engineers (IEEE). doi:10.1109/IEEESTD.2018.8332112 IEEE Std 1547-2018.
- [19] International Electrotechnical Commission (IEC). 2018. IEC 60076-7:2018: Power transformers – Part 7: Loading guide for mineral-oil-immersed transformers.
- [20] International Energy Agency (IEA). 2024. *Global EV Outlook 2024*. Technical Report. IEA, Paris. <https://www.iea.org/reports/global-ev-outlook-2024>
- [21] Zoe Juozapaitis et al. 2019. Explainable Reinforcement Learning via Reward Decomposition. In *IJCAI/ECAI Workshop on XAI*. IJCAI/ECAI Organization, Macau, China, 7 pages. <https://api.semanticscholar.org/CorpusID:204898543>
- [22] Arsenii Kuznetsov et al. 2020. Controlling Overestimation Bias with Truncated Mixture of Continuous Distributional Quantile Critics. arXiv:2005.04269
- [23] Sergey Levine et al. 2016. Learning Hand-Eye Coordination for Robotic Grasping with Deep Learning and Large-Scale Data Collection. *Int. J. Rob. Res.* 37 (2016), 421–436.
- [24] Timothy P. Lillicrap et al. 2019. Continuous control with deep reinforcement learning. arXiv:1509.02971 <https://arxiv.org/abs/1509.02971>
- [25] Haoye Lu et al. 2023. Multi-Objective Reinforcement Learning: Convexity, Stationarity and Pareto Optimality. In *11th Intl. Conf. on Learning Representations (ICLR)*. ICLR, Kigali, Rwanda, 27 pages.
- [26] Ingo Mauser et al. 2017. Definition, Modeling, and Communication of Flexibility in Smart Buildings and Smart Grid. In *Inter. ETG 2017*. VDE, Berlin, Germany, 1–6.
- [27] Steffen Meinecke et al. 2020. SimBench—A Benchmark Dataset of Electric Power Systems to Compare Innovative Solutions Based on Power Flow Analysis. *Energies* 13, 12 (Jun. 2020), 3290. doi:10.3390/en13123290
- [28] Volodymyr Mnih et al. 2016. Asynchronous Methods for Deep Reinforcement Learning. arXiv:1602.01783
- [29] netztransparenz.de. 2025. Spot Market Price according to § 3 No. 42a EEG. <https://www.netztransparenz.de> Accessed: 2025-07-01.
- [30] Lucas Nunes Alegre et al. 2022. MO-Gym: A Library of Multi-Objective Reinforcement Learning Environments. In *Proc. of the 34th Benelux Conf. on AI BNAIC/Benelearn 2022*. University of Antwerp, Antwerp, Belgium, 1–4. <https://bnaic2022.uantwerpen.be/>
- [31] Stavros Orfanoudakis et al. 2024. EV2Gym: A Flexible V2G Simulator for EV Smart Charging Research and Benchmarking. arXiv:2404.01849
- [32] Aisling Pigott et al. 2021. GridLearn: Multiagent Reinforcement Learning for Grid-Aware Building Energy Management. arXiv:2110.06396
- [33] Antonin Raffin et al. 2021. Stable-Baselines3: Reliable Reinforcement Learning Implementations. *Journal of Machine Learning Research* 22, 268 (2021), 1–8.
- [34] D. M. Roijers et al. 2013. A Survey of Multi-Objective Sequential Decision-Making. *Journal of Artificial Intelligence Research* 48 (Oct. 2013), 67–113.
- [35] John Schulman et al. 2017. Proximal Policy Optimization Algorithms. arXiv:1707.06347
- [36] John Schulman et al. 2017. Trust Region Policy Optimization. arXiv:1502.05477
- [37] David Silver et al. 2017. Mastering the game of Go without human knowledge. *Nature* 550, 7676 (2017), 354–359.
- [38] Matthijs T. J. Spaan. 2012. Partially Observable Markov Decision Processes. In *Reinforcement Learning: State-of-the-Art*. Springer, Berlin, Heidelberg, 387–414. doi:10.1007/978-3-642-27645-3_12
- [39] Tong Su et al. 2024. A Review of Safe Reinforcement Learning Methods for Modern Power Systems. <https://arxiv.org/abs/2407.00304>
- [40] R.S. Sutton and A.G. Barto. 2018. *Reinforcement Learning, second edition: An Introduction*. MIT Press, Cambridge, MA.
- [41] Harun Turker et al. 2014. Rule-Based Charging of Plug-in Electric Vehicles (PEVs). *IEEE Transactions on Power Delivery* 29, 3 (2014), 1012–1019.
- [42] U.S. Department of Energy. 2024. *The Future of Resource Adequacy*. Technical Report. U.S. Department of Energy. <https://www.energy.gov/sites/default/files/2024-04/2024%20The%20Future%20of%20Resource%20Adequacy%20Report.pdf>
- [43] Jose R Vazquez-Canteli et al. 2020. CityLearn: Standardizing Research in Multi-Agent Reinforcement Learning for Demand Response and Urban Energy Management. arXiv:2012.10504 <https://arxiv.org/abs/2012.10504>
- [44] Mnih Volodymyr et al. 2013. Playing Atari with Deep Reinforcement Learning. *CoRR* abs/1312.5602 (2013), 1–9. arXiv:1312.5602 <http://arxiv.org/abs/1312.5602>
- [45] Jianhong Wang et al. 2021. Multi-Agent Reinforcement Learning for Active Voltage Control on Power Distribution Networks. In *Advances in NIPS*, Vol. 34. Curran Associates, Inc., Red Hook, NY, USA, 3271–3284.
- [46] Christopher Yeh et al. 2024. SustainGym: Reinforcement Learning Environments for Sustainable Energy Systems. In *Advances in NIPS*, Vol. 36. Curran Associates, Inc., New Orleans, LA, USA, 59464–59476.
- [47] Peipei Yu et al. 2024. Safe Reinforcement Learning for Power System Control: A Review. arXiv:2407.00681 <https://arxiv.org/abs/2407.00681>
- [48] Chi Zhang et al. 2023. BEAR: Physics-Principled Building Environment for Control and Reinforcement Learning. In *Proc. 14th ACM Int. Conf. Future Energy Syst. (Orlando, FL, USA) (e-Energy '23)*. ACM, New York, NY, USA, 66–71.

A Appendix

A.1 Transformer Loading Guide Standards

Table 5 summarizes transformer overload limits according to IEEE C57.91 and IEC 60076-7. Transformers typically operate in daily 24-hour load cycles. For planned or cyclic overloads exceeding the rated capacity, temperature conditions and load duration must be considered to ensure safe operation. Transformers can tolerate short-term overloads of up to 200% of rated capacity for less than 30 minutes. Short-term overload capacity increases with cooler ambient temperatures or shorter durations. Larger transformers tolerate higher loads due to greater thermal inertia. Figure 7 shows transformer loading categories according to grid codes defined in Table 5. These include rated (100%, gray), cyclic (up to 150%, green), which typically occurs once per 24-hour load cycle, long-term (up to 180%, yellow) during extended fluctuations over a few hours, and short-term (up to 200%, red) for dynamic overloads. The axes represent active power P and reactive power Q , while the radial distance corresponds to the apparent power $S = \sqrt{P^2 + Q^2}$. While reactive power control supports voltage stability, it increases the reactive power flow at the PCC and distribution grid power exchange. This results in a higher apparent power of the transformer capacity and loadings. In eGridLVGym, grid operators can enforce constraints on transformer loading to mitigate such risks. When the apparent power exceeds a predefined threshold (typically the rated capacity), a hierarchical safety mechanism can optionally be activated. It first adjusts RL agent actions, prioritizing BES power reduction due to its fast and flexible response. Power exchange can be reduced by other measures such as PV curtailment, BEV, or HP control adjustments. Figure 6 depicts the algorithms behavior without transformer constraint enforcement. In contrast, Figure 8 shows enforced transformer constraints limiting loading to below 85%.

Table 5. Transformer Loading Guide Standards IEEE and IEC.

Operation Mode	IEEE C57.91 [17]	IEC 60076-7 [19]
Normal cyclic loading (24 hours)	110–120%	150%
Long-term loading (few hours)	130–150%	180%
Short-term loading (<30 min)	200%	200%

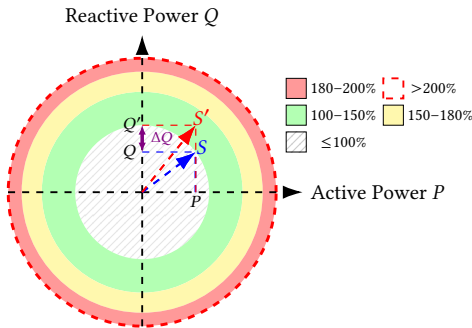


Fig. 7. Transformer apparent power diagram with concentric loading zones based on IEC 60076-7 (see Table 5).

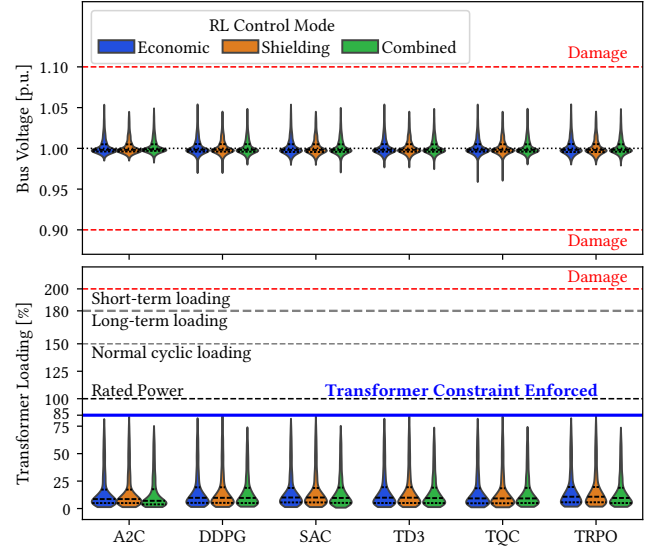


Fig. 8. RL modes performance in the eGridLV-feeder-v0 environment with the hard constraint safety mechanism activated. Top: Bus voltage levels (EN 50160). Bottom: Transformer loading limited to 85% rated capacity (IEC 60076-7).

A.2 Rule-Based Controller

This rule-based controller tracks a rolling average price:

$$c_k = \frac{1}{N_{24h}} \sum_{j=0}^{N_{24h}-1} \text{price}_{k-j}. \quad (\text{A.1})$$

At each timestep k , a decision variable $d_k \in \{-1, 0, 1\}$ is set as:

$$d_k = \begin{cases} 1, & \text{price}_k > 1.1 c_k, \\ -1, & \text{price}_k < 0.9 c_k, \\ 0, & \text{otherwise.} \end{cases} \quad (\text{A.2})$$

For each BES unit, the normalized action $u_k^{\text{BES}} \in [-1, 1]$ charges (< 0) or discharges (> 0):

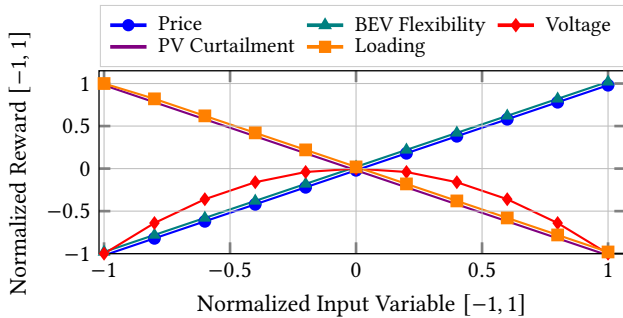
$$u_k^{\text{BES}} = \begin{cases} \max(-E_k^{\text{BES}}, -p_{\max}^{\text{BES}}) / p_{\max}^{\text{BES}}, & d_k = 1, \\ 0, & d_k = 0, \\ \min(E_{\max}^{\text{BES}} - E_k^{\text{BES}}, p_{\max}^{\text{BES}}) / p_{\max}^{\text{BES}}, & d_k = -1. \end{cases} \quad (\text{A.3})$$

For BEV systems, discharging is disabled below 75%, and half-power charging is used below 50% SoC level:

$$u_k^{\text{BEV}} = \begin{cases} \max(-E_k^{\text{BEV}}, -p_{\max}^{\text{BEV}}) / p_{\max}^{\text{BEV}}, & d_k = 1 \wedge \text{SoC} > 0.75, \\ \min(E_{\max}^{\text{BEV}} - E_k^{\text{BEV}}, p_{\max}^{\text{BEV}}) / p_{\max}^{\text{BEV}}, & d_k = -1, \\ 0.5, & \text{SoC} < 0.5, \\ 0, & \text{otherwise.} \end{cases} \quad (\text{A.4})$$

Table 6. Hyperparameters for SORL and MORL Algorithms.

Hyperparameter	Value
Common Settings	
Policy type	Multi-Layer Perceptron Policy
Number of parallel environments	4
Timesteps per rollout	672
Batch size	96
Evaluation frequency (episodes)	500
Number of episodes	8000
Total timesteps	5,376,000
PPO	
Neural network architecture	[64, 64]
Activation function	Rectified Linear Unit
Learning rate	0.002336
Entropy coefficient	0.01065
Discount factor (γ)	0.98
GAE lambda	0.95
Clip range	0.1
Max gradient norm	0.9
Number of epochs	1
TRPO	
Num of steps per update / Batch size	672 / 96
SAC/DDPG/TD3	
Learning starts	96
Recurrent PPO	
Policy type	Multi-Layer Perceptron Long Short-Term Memory Policy
Num of steps per update / Batch size	96 / 96
CAPQL	
Learning starts	96
Batch size	96
Number of episodes	8000
Reference point	[4, -2]

Fig. 9. Normalized reward functions based on scaled input variables. All components are projected to $[-1, 1]$ to ensure consistent reward values and stable learning across tasks.

For HP elements simply invert d_k :

$$u_k^{\text{HP}} = -d_k. \quad (\text{A.5})$$

This approach increases energy consumption when prices are low and decreases when prices are high by shifting the HP's originally planned consumption profile. This also considers the net heat consumption deviation from Eq. (5d).

A.3 Sensor Modeling

Measurement noise is introduced via a positively truncated Gaussian term, $x_{\text{noise}} \sim \text{TruncNorm}(a = 0, b = 1, \mu = 0, \sigma = \sigma_{\text{orig}})$. By setting $\mu = 0$ (before truncation), the noise is symmetric around zero but constrained to $[0, 1]$, thus avoiding physically implausible negative PV values at night. The parameter σ_{orig} corresponds to the empirical standard deviation derived from the original data. Consequently, the virtual measurement value is

$$x_{\text{measurement}} = x_{\text{orig}} + x_{\text{noise}}, \quad (\text{20})$$

where x_{noise} denotes the positively truncated Gaussian noise term.

A.4 Hyperparameter Settings

The default hyperparameters for the SORL algorithms (DPG, TD3, SAC, TQC, A2C, TRPO, and PPO) and MORL algorithm (CAPQL) are listed in Table 6.

A.5 Reward Design

This section explores in more detail the design of the reward function Eqs. (11–17) concerning its interpretability and informativeness, which are essential for designing effective and efficient RL systems. The soft penalties in Eqs. (16–17) guide the agent to learn a control policy that respects physical constraints while optimizing the objectives. These soft constraints are overridden by hard constraints under physically infeasible operating conditions. This ensures strict enforcement of physical and grid constraint operation. Linear reward functions for price signals and energy flexibility are based on normalized inputs, ensuring interpretable and informative rewards. An identical approach applies to the BEV flexibility and curtailment components. Stricter penalties can be imposed for overload violations. However, high penalties can negatively affect policy learning and lead to suboptimal performance. A triangular penalty function can also be used, which penalizes deviations linearly in contrast to the quadratic approach. This could lead to a less aggressive reaction to significant voltage fluctuations. In contrast, quadratic rewards provide tolerance for minor voltage deviations by penalizing small errors less severely. This effectively prevents oscillations and repeated activation-deactivation cycles. A piecewise deadband function, similar to a quadratic penalty, can also be used. Reward components provide transparency by directly linking to physical and economic parameters (e.g., the price reward, which offers an intuitive measure of costs). Similarly, they ensure the informativeness of the reward function by providing both measurement and prediction-based feedback at each time step, accelerating the learning process, and providing immediate insights even for minor changes. Figure 9 illustrates the normalized reward functions used for different task components during the learning process, applied consistently for both the SORL and MORL agents.