Reinforcement Learning Control of MicroGrid Systems

Farooq Olanrewaju (g202404900)[1], Abubakar Abdulkarim (g202421140)[2]
[1]*Control & Instrumentation Eng. Dept.*, *King Fahd Univ. of Petroleum & Minerals*, Dhahran 31261, Saudi Arabia
[2]*Electrical Eng. Dept.*, *King Fahd Univ. of Petroleum & Minerals*, Dhahran 31261, Saudi Arabia

***ABSTRACT*—Microgrids combine renewable generation, dispatchable resources, energy storage, and local loads, and can operate either grid-connected or islanded. Rapid renewable intermittency, stochastic demand, and time-varying electricity prices make real-time energy management challenging for fixed-rule or strictly model-based controllers, particularly when reliability constraints must be respected. This paper proposes a reinforcement-learning (RL) energy management system (EMS) for a hybrid microgrid comprising photovoltaic and wind generation, battery storage, a dispatchable unit, and grid import/export. The control problem is cast as a continuous Markov decision process and implemented in an OpenAI Gym-compatible simulator driven by load and resource time-series (measured or synthetically generated). We benchmark a rule-based EMS against DRL agents trained with Proximal Policy Optimization (PPO), Twin Delayed Deep Deterministic Policy Gradient (TD3), and Soft Actor-Critic (SAC). Across the studied scenarios, unstructured (random) actions produce frequent power-balance violations and load shedding, whereas learned continuous-control policies improve supply adequacy and reduce unmet demand and renewable curtailment while coordinating storage and grid trading within operational limits. The study also highlights the sensitivity of learned performance to environment fidelity, motivating future extensions that explicitly model degradation and outage/repair processes for deployment-ready evaluation.**

***Index Terms*—Microgrid, energy management system, reinforcement learning, deep reinforcement learning, continuous control, PPO, TD3, SAC, grid trading, reliability.**

I. Introduction

A microgrid is a controllable, localized portion of the distribution network that can operate connected to the main grid or in islanded mode [**?**], [**?**]. Microgrids typically integrate renewable generation mostly photovoltaic (PV) and wind, dispatchable generation, energy storage, and diverse loads both residential and industrial as shown in Fig.1. An energy management system (EMS) coordinates these assets to minimize operating cost while meeting reliability and power-quality requirements. However, increasing renewable penetration introduces fast variability and uncertainty; furthermore, component failures, grid outages, and time-varying electricity prices complicate optimal control and can degrade supply security if not handled explicitly [**?**].

Conventional EMS approaches include rule-based heuristics, dynamic programming, and model predictive control (MPC). These methods provide structured constraint handling, but typically require explicit models and forecasts; their performance may deteriorate under significant uncertainty, modeling mismatch, or rare-event contingencies. Motivated by the need for adaptive decision-making under uncertainty, recent work has explored reinforcement learning (RL) for microgrid energy management [**?**]. In parallel, open-source simulators such as `pymgrid` have lowered the barrier for RL-oriented EMS research by standardizing environments and interfaces for training and evaluation [**?**]. Nevertheless, many published RL studies employ simplified discrete states/actions and often omit practical phenomena such as battery degradation, repair costs, and multi-load interactions, which can materially change optimal operating strategies and the realism of performance claims.

This paper develops a continuous-control RL EMS that integrates local generation, storage, and grid trading to optimize cost-efficiency, reliability, and resilience under variable domestic and industrial loads. The EMS is formulated as a Markov decision process (MDP) with continuous observations and actions, enabling direct learning of practical setpoints using modern continuous-control DRL algorithms such as PPO, TD3, and SAC [**?**], [**?**], [**?**].

Figure 1. Microgrid system overview

A. Contributions

The main contributions are:

- A continuous-control MDP formulation for hybrid microgrid EMS with renewables, storage, dispatchable genera-

tion, and grid trading.
- A comparative analysis of modern DRL controllers (PPO/TD3/SAC) against rule-based baselines under variable residential and industrial demand.
- Development of a Python-based microgrid simulation and Gym-style environment built around realistic PV, wind, battery, grid, and failure models.

B. Paper organization

Section II reviews related work on optimization/MPC-based EMS and RL-based EMS, including safety and benchmarking considerations. Section III presents the microgrid model and MDP formulation. Section IV describes the learning algorithms and experimental methodology. Section V reports results and discussion. Section VI concludes and outlines future work.

II. Literature review

Microgrid EMS research broadly spans (i) optimization- and MPC-based methods that use explicit models and forecasts, and (ii) learning-based methods, particularly RL/DRL, that learn policies from interaction. Across both categories, the core challenge is balancing economics and reliability under uncertainty while respecting device and network constraints [**?**].

A. Optimization- and MPC-based EMS under uncertainty

Optimization-based EMS methods (deterministic, stochastic, or robust) are attractive because they encode constraints directly and can incorporate tariffs, reserves, and operational priorities. MPC extends this framework by repeatedly solving a constrained optimization problem over a receding horizon, enabling feedback through re-optimization as forecasts update. However, MPC performance depends strongly on model fidelity and forecast quality, and it can be stressed by high renewable variability, heterogeneous loads, and contingencies such as islanding and component failures. Moreover, multi-timescale operation like day-ahead planning plus real-time correction raises coordination questions; approaches that couple operational planning and real-time optimization via value-function or cost-to-go ideas attempt to address this gap by embedding longer-horizon consequences into real-time decisions [**?**]. These limitations motivate adaptive strategies that can react effectively even when explicit models are imperfect.

B. RL/DRL for microgrid energy management

RL formulates EMS as sequential decision-making, learning a policy that maps states to actions to maximize long-term return. Early work showed that RL can handle stochastic renewables and demand and can learn effective EMS policies without an explicit transition model [**?**]. As EMS models grow in dimensionality and nonlinearity, DRL becomes important; empirical studies comparing DRL algorithms for microgrid EMS with flexible demand report that algorithm choice and training stability significantly affect convergence and operational performance [**?**]. Even with this promising results, many EMS-RL studies simplify action spaces like discrete charge/discharge modes and omit important operational mechanisms such as outages/repairs and degradation, which can inflate performance estimates and reduce transferability.

C. Continuous-control DRL and actor–critic methods

Practical EMS decisions are often continuous (battery power, grid import/export, dispatchable generation). Continuous-control DRL is therefore a natural fit, avoiding coarse discretization that can reduce optimality and induce switching behavior near constraints. Modern actor–critic methods are widely used in continuous control: PPO stabilizes policy-gradient updates using clipped objectives [**?**], TD3 reduces overestimation bias using twin critics and delayed updates [**?**], and SAC optimizes a maximum-entropy objective to encourage exploration and improve robustness [**?**]. For EMS, these properties are relevant because the environment is non-stationary and may include rare but consequential events such as grid outages.

D. Safety, constraints, and feasibility in EMS-RL

EMS operation is safety-critical: actions must respect SOC bounds, power limits, and import/export capacities while maintaining supply to critical loads. Standard RL exploration does not guarantee constraint satisfaction, motivating safe and constrained RL. Survey work categorizes safe RL methods into approaches that modify the optimality criterion like risk-sensitive or constrained objectives and those that incorporate external knowledge such as shielding, action projection, or safety filters [**?**]. CMDP-based methods provide a principled framework by treating constraint violations as separate cost signals; constrained policy optimization is a representative approach that updates policies while enforcing constraints under trust-region style bounds [**?**]. In practice, EMS-RL studies often combine DRL with engineering safeguards to ensure physical feasibility during training and deployment.

E. Battery degradation and lifecycle-aware EMS

Battery storage is central to microgrid flexibility, but cycling accelerates degradation and changes the true economic optimum. Many EMS-RL studies treat storage as an ideal buffer with fixed capacity, which can overstate savings and lead to unrealistic dispatch patterns. Battery aging mechanisms and their dependence on operating conditions are well documented [**?**], motivating degradation-aware EMS that includes lifecycle costs or proxy penalties, throughput- or SOC-swing-based terms in the objective. From an RL point, adding degradation costs changes the reward landscape and can shift learned policies toward gentler cycling; however, the lack of standardized degradation models and evaluation protocols remains a key barrier to cross-paper comparability.

F. Benchmarking and open simulation environments

Because EMS-RL outcomes depend heavily on environment design and evaluation protocol, benchmarking and reproducibility are recurring concerns. Open-source environments support more credible comparisons by standardizing interfaces and scenarios. `pymgrid` provides an RL-oriented microgrid simulator aimed

at tertiary EMS research [**?**], while OpenModelica Microgrid Gym offers a Gym-compatible environment for microgrid control experimentation [**?**]. Despite these advances, unified benchmarks that simultaneously capture continuous-control EMS setpoints, explicit outage/repair processes, degradation-aware storage modeling, and multi-load reliability metrics remain limited.

III. Methodology

This section presents the component-level mathematical models used to simulate the microgrid dynamics and defines the learning-based energy management strategy. The overall goal is to obtain a tractable yet physically meaningful environment where an RL agent can learn continuous setpoints for storage and grid exchange while respecting operational limits and reliability objectives.

A. Mathematical Models

We model the microgrid as a discrete-time system with time step Δt. At each step, renewable generation and loads are treated as exogenous inputs (measured or time-series driven), while the controllable assets (battery and grid exchange) are actuated by the EMS.

1) Photovoltaic (PV) Model

The PV generation is computed using a commonly adopted irradiance–temperature performance model [**?**], [**?**]. The PV power output is given by

$$P = P_r \mu \frac{G}{G_{ref}} \left[1 + \gamma \left(T_{cell} - T_a\right)\right] \tag{1}$$

where P is the PV power output (kW), P_r is the rated PV power (kW), μ is a derating factor accounting for aggregate non-idealities (e.g., soiling, wiring losses, mismatch, and inverter losses), G is solar irradiance, and G_{ref} is the PV reference irradiance (typically standard test conditions). The bracketed term captures the first-order sensitivity of PV output to temperature: γ is the temperature coefficient, T_{cell} is the PV cell temperature, and T_a is the ambient temperature. This model provides a lightweight but effective mapping from weather inputs to available PV power, making it suitable for control-oriented simulation and RL training where many rollouts are required [**?**], [**?**].

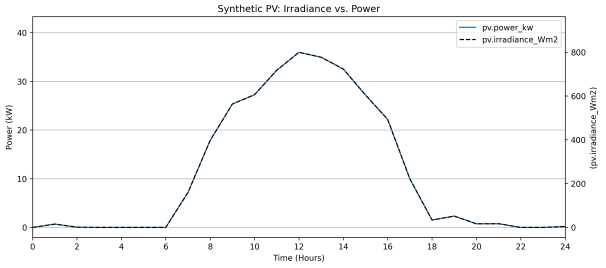

Figure 2.

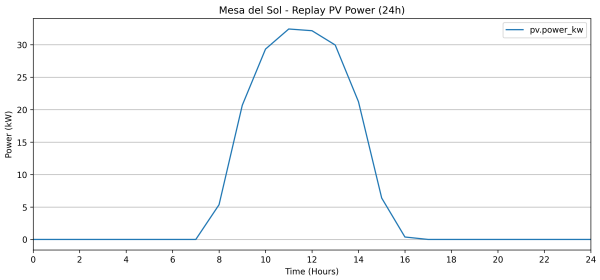

Figure 3.

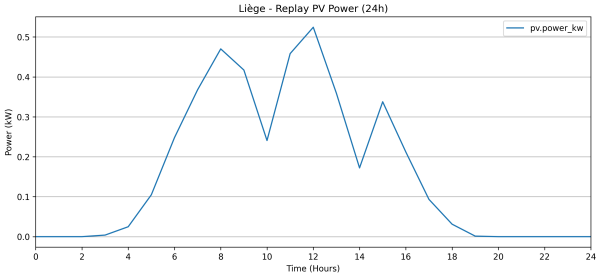

Figure 4.

2) Wind Turbine Model

Wind generation is modeled using a piecewise power curve parameterized by cut-in, rated, and cut-out wind speeds [**?**]. The wind turbine output is defined as

$$P = \begin{cases} 0 & \text{if } v < v_{ci} \\ P_r \frac{v - v_{ci}}{v_r - v_{ci}} \Delta t & \text{if } v_{ci} \le v < v_r \\ P_r \Delta t & \text{if } v_r \le v < v_{co} \\ 0 & \text{if } v > v_{co} \end{cases} \tag{2}$$

where v is the current wind speed (m/s), v_{ci}, v_r, and v_{co} are the cut-in, rated, and cut-out speeds (m/s), P_r is the rated wind turbine power (kW), and Δt is the time interval (s). The piecewise structure captures the physical operating regimes: no production at low wind speeds, a ramp-up region as aerodynamic power increases, a rated plateau due to generator/controls saturation, and shutdown at extreme winds for protection [**?**]. In our simulator, this model provides the available wind contribution to the instantaneous power balance at each step.

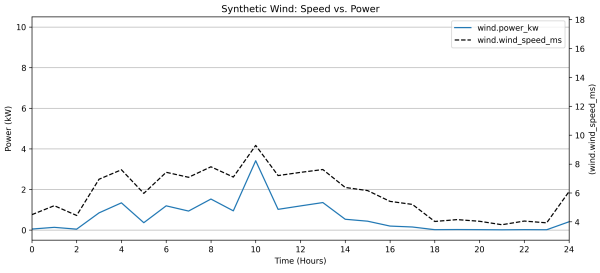

Figure 5.

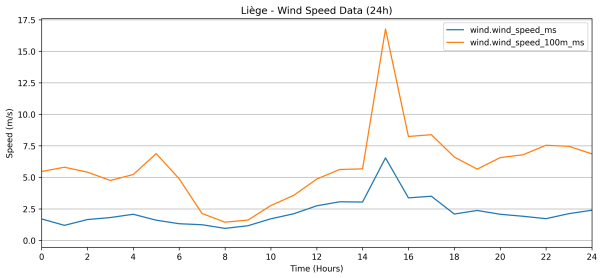

Figure 6.

3) Battery Model

Battery dynamics are represented through the state of charge (SOC) and a simplified capacity aging model [**?**]. The SOC is defined as:

$$SOC(t) = \frac{C_s(t)}{C_n(t)} \tag{3}$$

where $C_s(t)$ is the current stored charge (C) and $C_n(t)$ is the nominal charge storage capacity (C). The SOC update under charging and discharging is modeled as

$$SOC(t+1) = SOC(t) + \mu_c \frac{P\Delta t}{C_s} \quad \text{(Charging)} \tag{4}$$

$$SOC(t+1) = SOC(t) - \mu_d \frac{P\Delta t}{C_s} \quad \text{(Discharging)} \tag{5}$$

where P is the battery power (kW) applied during Δt and μ_c and μ_d are charging/discharging coefficients that represent conversion losses. Operationally, $SOC(t)$ is constrained to remain within allowable limits (e.g., $SOC_{\min}$ and $SOC_{\max}$), and power setpoints are saturated to respect charge/discharge limits.

To reflect long-term performance degradation, the nominal capacity is updated using a throughput/SOC-swing degradation proxy [**?**]:

$$C_n(t) = C_n(t-1) - C_n(0)\,\varphi\,[SOC(t-1) - SOC(t)] \tag{6}$$

where φ is the aging coefficient. Although simplified compared to electrochemical aging models, this formulation introduces an explicit coupling between cycling behavior and usable capacity, enabling evaluation of control policies under non-ideal storage evolution [**?**].

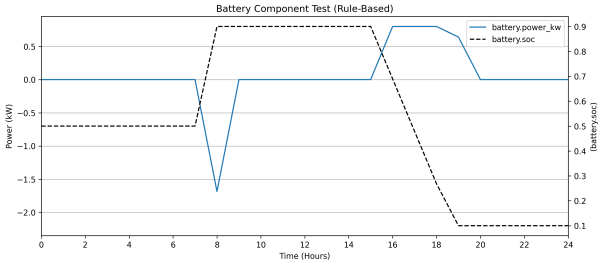

Figure 7.

4) Grid Model and Power Balance

At each time step, the microgrid enforces power balance between supply and demand. We define aggregate generated and consumed power as

$$\sum P_{gen} = \sum P_{PV} + P_W + P_{BC} + P_{GS} \tag{7}$$

$$\sum P_{con} = \sum P_F + P_R + P_{BD} + P_{GB} \tag{8}$$

and the net power as

$$P_{net} = \sum P_{gen} - \sum P_{con}. \tag{9}$$

Here, P_{PV} and P_W are the PV and wind contributions, P_R and P_F denote residential and factory loads, P_{BC} and P_{BD} are battery charge/discharge powers, and P_{GS} and P_{GB} represent grid export (sell) and import (buy), respectively. The sign conventions are chosen such that $P_{net} \geq 0$ indicates a surplus (exporting is possible), while $P_{net} < 0$ indicates a deficit (importing is required).

The grid is treated as a slack bus with finite import/export capacities. When surplus power exists, export is limited by P_{GI} (maximum grid injection). If P_{net} exceeds this limit, excess generation must be curtailed:

$$P_{curt} = P_{net} - P_{GI} \quad \text{if } P_{net} > P_{GI}. \tag{10}$$

When a deficit exists, import is limited by P_{GE} (maximum grid extraction). If the deficit exceeds the import capability, unmet demand occurs:

$$P_{unmet} = P_{GE} - P_{net} \quad \text{if } P_{net} < P_{GE}. \tag{11}$$

The resulting quantities P_{curt} and P_{unmet} are key reliability/efficiency indicators and are explicitly penalized in the RL reward design.

B. Failure Modelling

To evaluate controller robustness under disturbances and rare events, we employ a time-varying failure rate model. The failure parameter θ is modulated by external conditions as

$$\theta = \theta_{base}\,[1 + \omega \max(0,\, \theta_{ext} - \theta_{thresh})] \tag{12}$$

where θ_{base} is the baseline failure rate, θ_{ext} is an external stress indicator, θ_{thresh} is a threshold above which external stress increases failure propensity, and ω controls sensitivity to externally induced failure.

Assuming a Poisson failure process, the probability of failure occurrence during an interval Δt is

$$P = 1 - e^{-\theta \Delta t}. \tag{13}$$

Following a failure event, the time-to-recover T is modeled as exponentially distributed with mean $MTTR$:

$$T \sim \text{Exp}(\lambda), \quad \lambda = \frac{1}{MTTR}. \tag{14}$$

This yields a parsimonious outage/repair mechanism that can be integrated into simulation rollouts to stress-test learned policies under reduced availability or islanded-like conditions.

C. Reinforcement Learning Controller

The EMS is formulated as a Markov decision process (MDP) in which the agent observes a state s_t and selects an action a_t at each time step to maximize expected discounted return. The state vector aggregates information required for real-time decisions, including (but not limited to) load levels (P_R, P_F), renewable availability (P_{PV}, P_W or their exogenous drivers), battery SOC and operational limits, and grid exchange limits. The action space is continuous and corresponds to real-valued setpoints for controllable power flows (e.g., battery charge/discharge command and/or grid import/export scheduling subject to saturation and SOC feasibility).

We implement the RL controller using the Stable-Baselines3 toolbox with actor–critic function approximation and a `MultiInputPolicy` to accommodate multi-field observations. We evaluate multiple widely used DRL algorithms—PPO, A2C, TD3, and SAC—using their standard library implementations [**?**], [**?**], [**?**]. For each method, separate neural networks are learned for the policy (actor) and value estimation (critic), and action outputs are clipped/projection-filtered to ensure physical feasibility (SOC bounds, power limits, and grid exchange constraints). Training is performed by rolling out the simulator over representative time-series scenarios and updating the policy to improve long-horizon performance.

D. Reward Function

The reward at each time step is designed to encode economic operation while strongly discouraging reliability violations and renewable wastage. The implemented reward is a weighted combination of four terms:

$$r_t = w_{\text{cost}}\, c_t - w_{\text{unmet}}\, u_t - w_{\text{curt}}\, x_t - w_{\text{soc}}\, d_t \tag{15}$$

where c_t is the step-wise economic term (computed from energy purchasing/selling and any operational costs in the simulator), u_t is unmet demand (e.g., P_{unmet}), x_t is curtailed energy/power (e.g., P_{curt}), and d_t is an optional SOC deviation penalty.

In our implementation, the weights are

$$\begin{aligned} w_{\text{cost}} &= 5.0 \\ w_{\text{unmet}} &= 10.0 \times 3.5 = 35.0 \\ w_{\text{curt}} &= 0.1 \times 1.5 = 0.15 \\ w_{\text{soc}} &= 0.0. \end{aligned} \tag{16}$$

This choice places the highest priority on reliability (large penalty on unmet demand), followed by economic performance, with a smaller but nonzero penalty on renewable curtailment. The SOC deviation term is disabled in the present experiments ($w_{\text{soc}} = 0$), but it can be activated in future work to encourage SOC regularization (e.g., to reduce deep cycling or maintain reserve capacity).

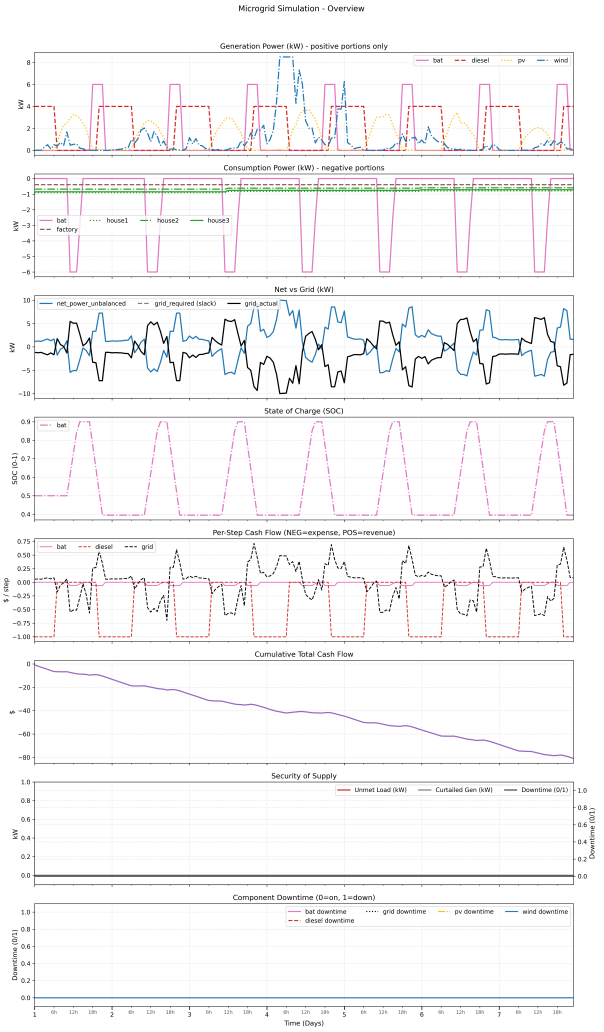

Figure 8.

IV. Results

A. Rule-Based Control with Synthetic Data

B. Rule-Based Control with Real Datasets

C. RL Control

D. PPO

E. A2C

F. SAC

G. TD3

V. Conclusion and Future Work

This paper presented a continuous control RL EMS for a hybrid microgrid integrating photovoltaic and wind generation, battery energy storage, a dispatchable generator, and grid import/export. The EMS problem was formulated as a continuous Markov decision process and implemented in an OpenAI Gym-compatible simulation environment that enforces operational constraints like state-of-charge bounds, charge/discharge limits, and grid trading limits while capturing time-varying renewable availability and heterogeneous residential and industrial demand. Using a rule-based controller as a baseline, we evaluated

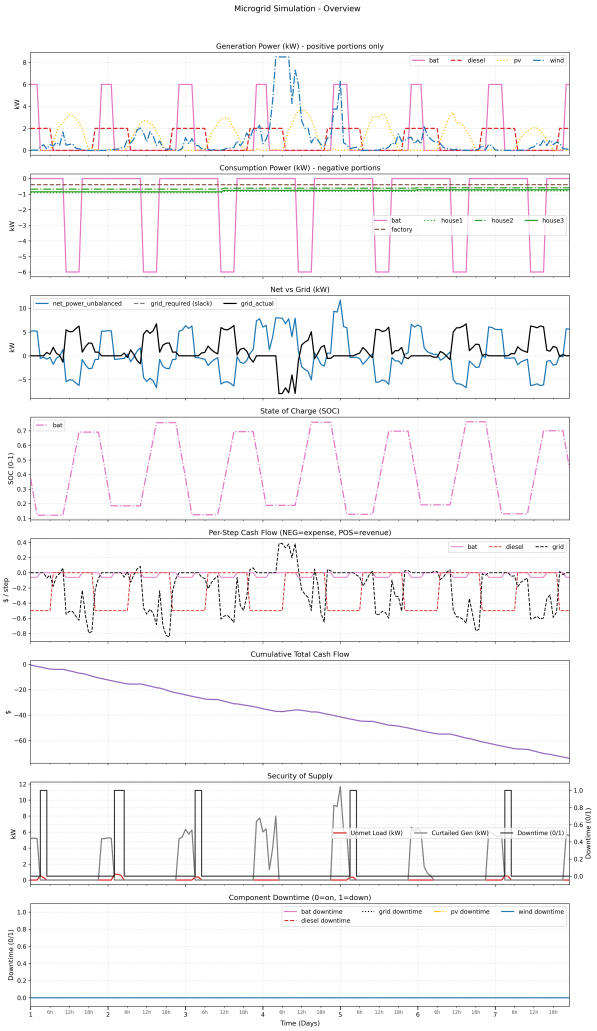

Figure 9.

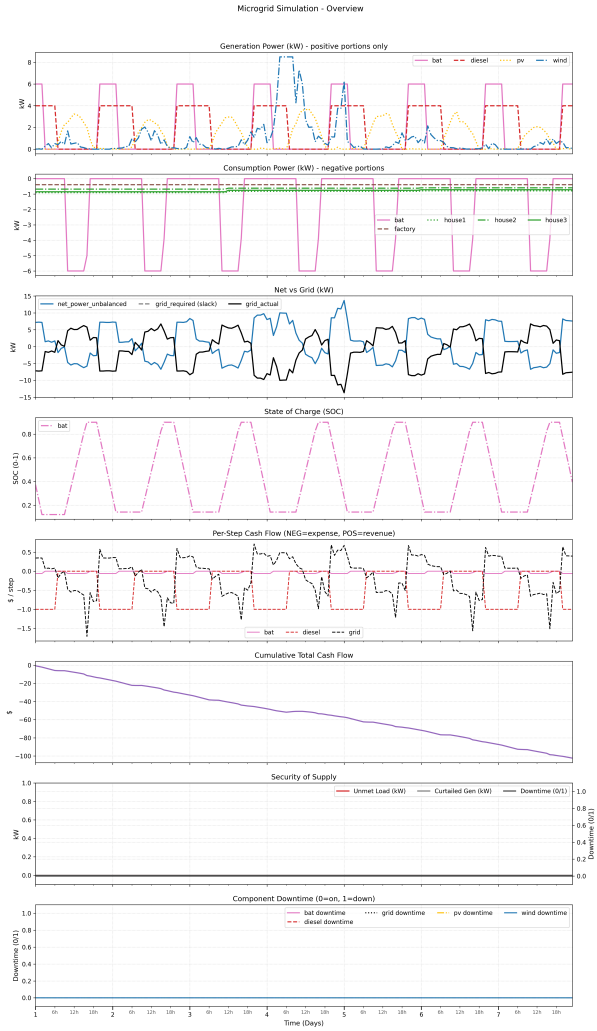

Figure 10.

modern continuous-control DRL algorithms PPO, TD3, and SAC to quantify their capability to learn coordinated storage dispatch and grid-trading actions.

The results demonstrate that random actions policies produce frequent power-balance violations, unmet load events, and unstable operation, underscoring the difficulty of microgrid EMS under uncertainty. In contrast, trained RL policies improve supply adequacy and sustain power balance more consistently in the studied scenarios by learning to allocate renewable generation, storage charging/discharging, and grid import/export in a coordinated manner. The comparative analysis also indicates that algorithm choice influences policy smoothness and operational trade-offs, and that observed performance is sensitive to environment assumptions, reward shaping, and the fidelity of component models. These findings support the feasibility of continuous control RL for microgrid EMS while highlighting the need for careful benchmarking and realism to ensure deployment relevance.

Future work will focus on increasing modeling fidelity and strengthening evaluation rigor. First, we will incorporate richer contingency modeling, including component-specific failure modes, external-stress dependent outage rates, repair costs, and stochastic repair-time distributions, enabling resilience oriented training and assessment. Second, we will integrate degradation-aware storage models that capture cycle aging and calendar aging effects, allowing the EMS to explicitly optimize lifecycle cost rather than short-term energy arbitrage alone. Third, we will tighten coupling between the EMS layer and lower-level microgrid dynamics by interfacing with voltage/frequency control and converter constraints, enabling evaluation of how EMS decisions interact with stability margins during both grid-connected and islanded transitions.

VI. Data and Code Availability

All source code, simulation environments, and supporting materials for this work are publicly available in the GitHub repository titled microgrid-control-sim[1]. The repository includes a simple `README` file with instructions, enabling full reproduction of the experimental results and facilitating adaptation for future research projects.

[1]Link: https://github.com/olanrewajufarooq/microgrid-control-sim

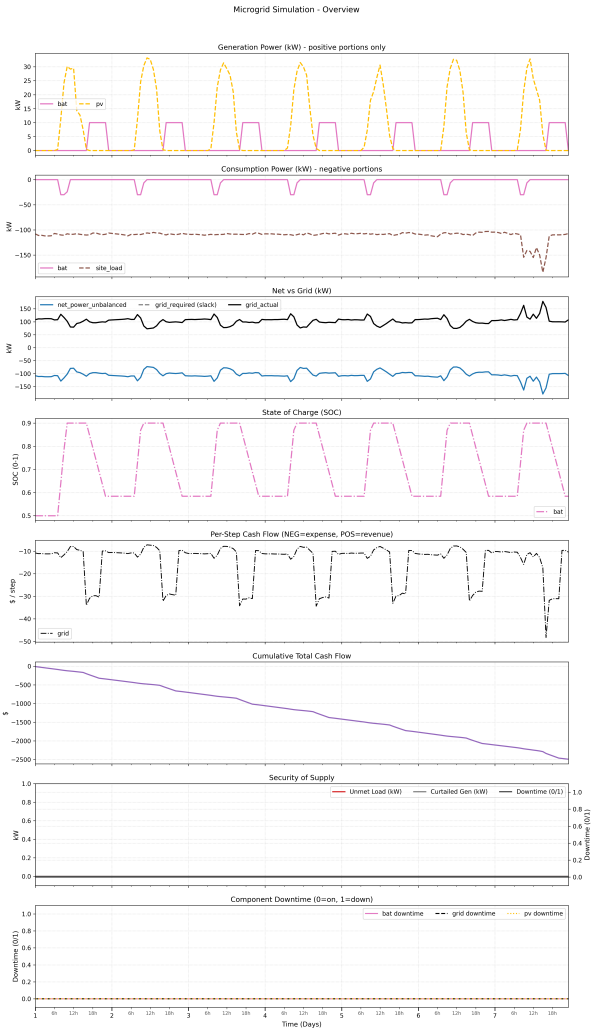

Figure 11.

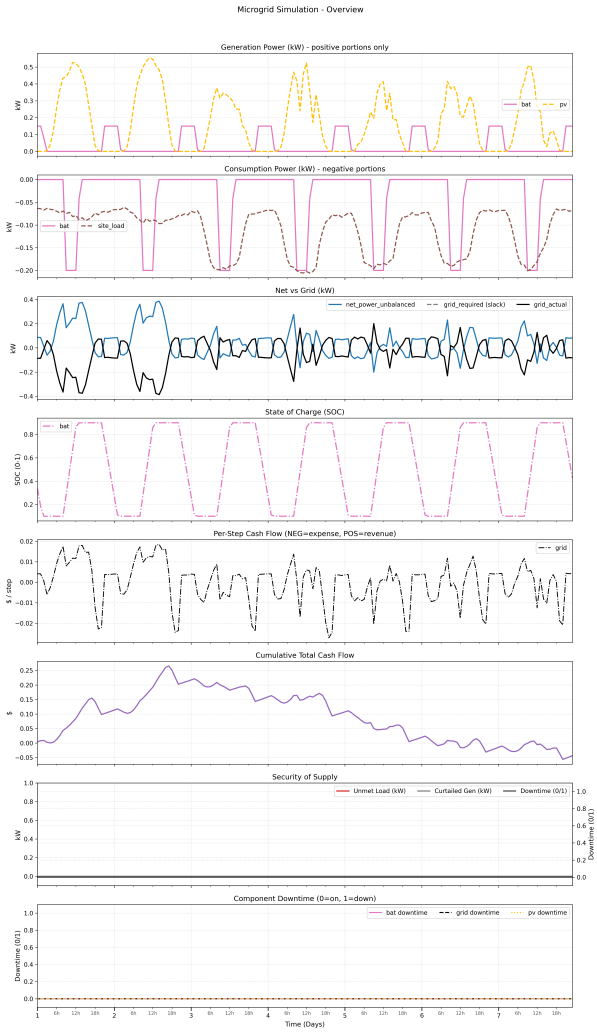

Figure 12.

Figure 13.

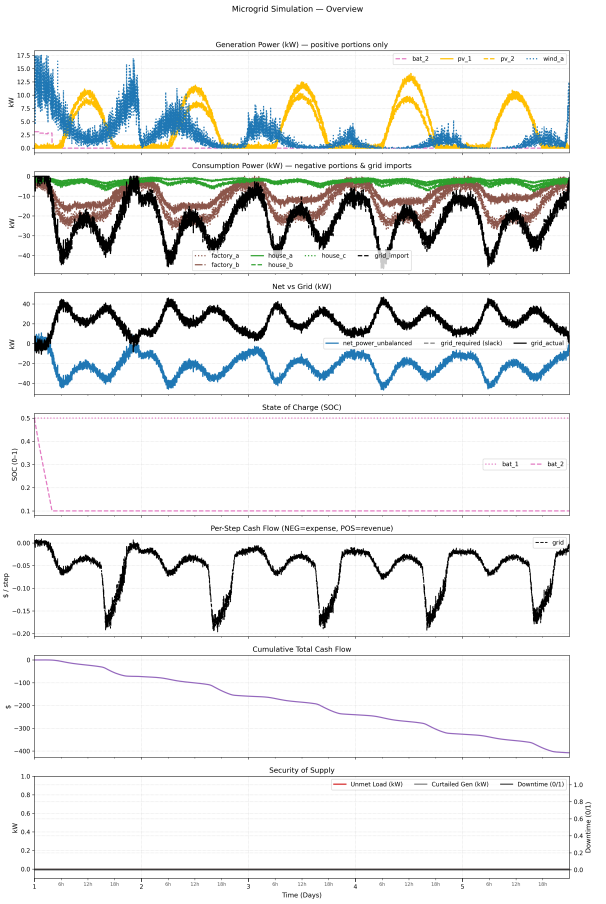

Figure 14.

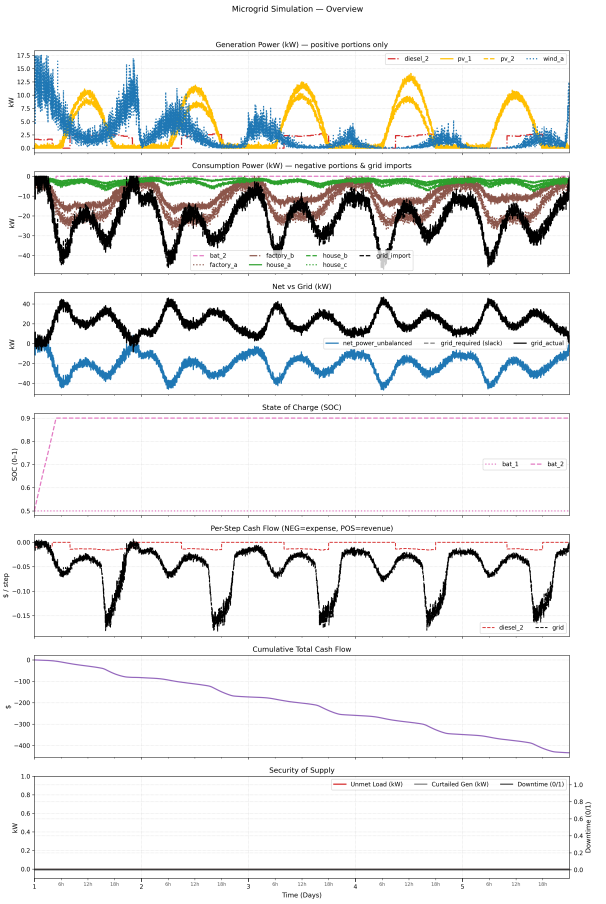

Figure 16.

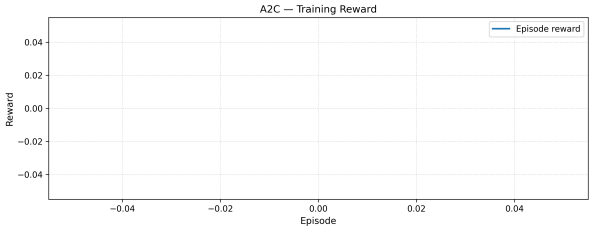

Figure 15.

Figure 17.

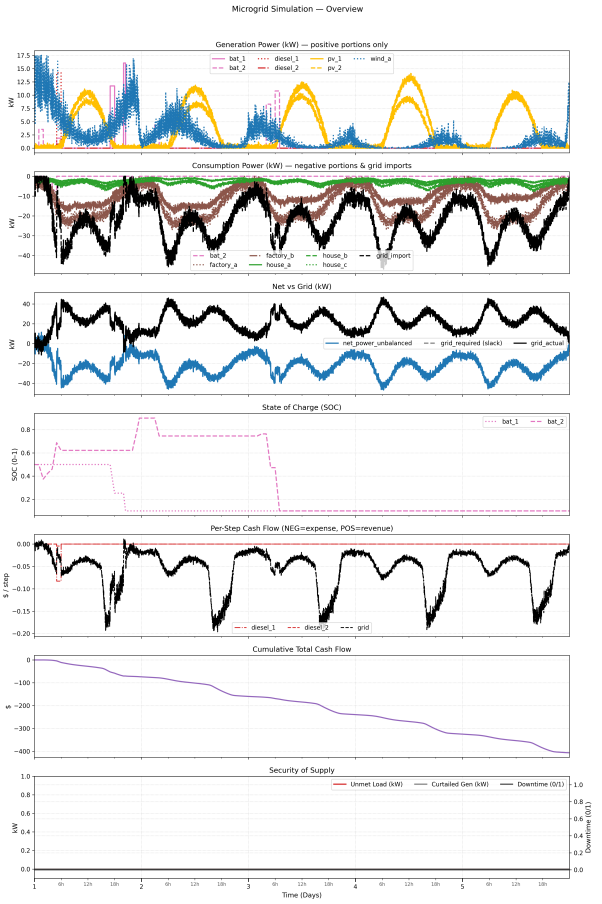

Figure 18.

Figure 19.

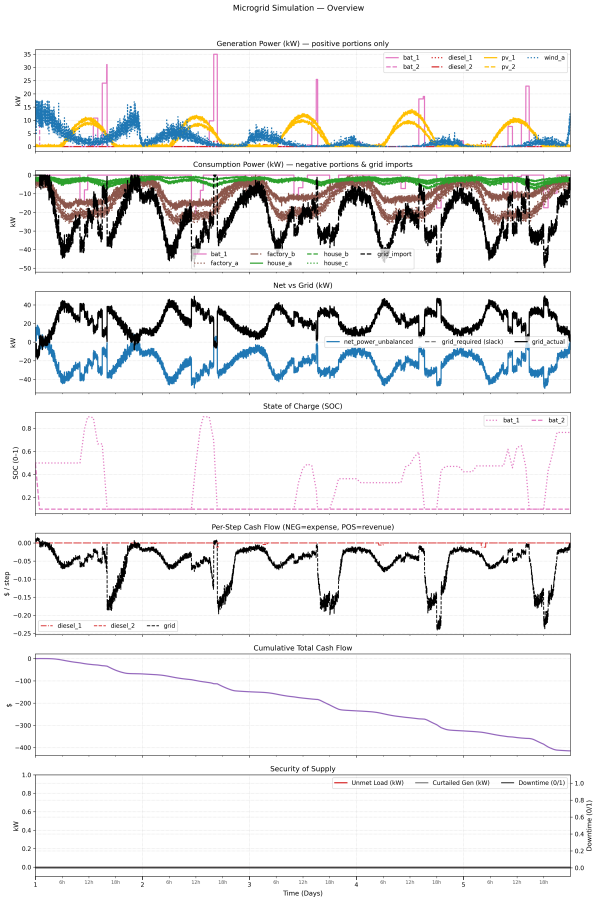

Figure 20.