Optimizing Building Energy Management using Offline and Online Reinforcement Learning: A Comparative Study of CQL, SAC, and TD3

Mohamed Stifi [*1] **Carlos Adonon** [*1] **Hafssa Bounia** [*1]

Abstract

Building Energy Management Systems (BEMS) are critical for reducing global energy consumption while maintaining occupant comfort. In this work, we present a comparative study of Reinforcement Learning (RL) algorithms for controlling HVAC and battery storage in a 5-zone building using the Sinergym simulator. We evaluate Conservative Q-Learning (CQL) for offline learning, and Soft Actor-Critic (SAC) and Twin Delayed DDPG (TD3) for online learning. Our results show that TD3 significantly outperforms other agents in terms of total reward ($-\mathbf{68.46}$), successfully balancing thermal comfort constraints. However, we observe a distinct trade-off: TD3 incurs higher electricity costs ($62.32) compared to SAC and CQL (32$), which converged to passive policies that minimize cost at the expense of extreme comfort violations.

1. Introduction

The building sector accounts for nearly 40% of global energy consumption, necessitating intelligent control systems to optimize efficiency. Traditional rule-based controllers (RBC) often lack the flexibility to adapt to dynamic electricity pricing and intermittent renewable energy sources like Solar PV. Reinforcement Learning (RL) has emerged as a powerful tool for optimal control in this domain (Mason & Gdomb, 2019).

This paper addresses the problem of optimizing a building's HVAC and battery systems to minimize electricity costs while maintaining thermal comfort. We investigate two paradigms: *Online RL*, where agents interact directly with the simulator, and *Offline RL*, where agents learn from a fixed dataset to mitigate safety risks associated with exploration.

Our contributions are:

- A custom Gym wrapper integrating battery dynamics, solar production, and dynamic pricing into *Sinergym*.
- A rigorous comparison of SAC, TD3, and CQL based on energy cost, thermal comfort, and hardware health.
- An analysis of the ”cost vs. comfort” trade-off, highlighting how different algorithms prioritize conflicting objectives.

2. Background

2.1. Online Algorithms

Soft Actor-Critic (SAC): An off-policy algorithm that maximizes a trade-off between expected return and entropy (Haarnoja et al., 2018). It is favored for its sample efficiency and stability.

Twin Delayed DDPG (TD3): An extension of DDPG that addresses Q-value overestimation by using two critic networks and taking the minimum Q-value, along with delayed policy updates (Fujimoto et al., 2018).

2.2. Offline Algorithms

Conservative Q-Learning (CQL): Offline RL aims to learn policies from static datasets without environment interaction. CQL learns a conservative Q-function lower bound to prevent overestimation of out-of-distribution actions, a common failure mode in offline RL (Kumar et al., 2020).

3. Problem Setup

3.1. Environment

We utilize **Sinergym** (Jiménez-Raboso et al., 2021), specifically the `Eplus-5zone-hot-continuous-v1` environment. We wrap this environment to include:

- **Battery:** 10 kWh capacity, 90% efficiency.
- **Solar PV:** 5 kW peak power.
- **Dynamic Pricing:** On-peak (0.25 €/kWh) and Off-peak (0.10 €/kWh) rates.

[*] Equal contribution [1] major intelligence artificielle engineering at Ensias, Supervised by Mohamed Naoum.

3.2. MDP Formulation

State Space ($\mathcal{S} \in \mathbb{R}^{20}$): Includes zone temperatures, outdoor weather (temperature, humidity, solar irradiance), current power load, time of day, and battery State of Charge (SOC).

Action Space ($\mathcal{A} \in \mathbb{R}^{3}$): Continuous control over:

1. Heating Setpoint ($12°\text{C} - 23.25°\text{C}$).
2. Cooling Setpoint ($23.25°\text{C} - 30°\text{C}$).
3. Battery Flow (-1 discharge to $+1$ charge).

Reward Function: We design a multi-objective reward to balance conflicting goals:

$$r_t = -C_{elec} - \lambda_c P_{comfort} - \lambda_b P_{battery} \tag{1}$$

Where C_{elec} is the electricity cost, $P_{comfort}$ is a quadratic penalty for temperature violations outside $[21°\text{C}, 25°\text{C}]$, and $P_{battery}$ penalizes battery cycling. We set $\lambda_c = 1.0$ and $\lambda_b = 0.1$.

4. Methodology

4.1. Agent Architecture

All three agents (CQL, SAC, TD3) utilize neural networks with 2 hidden layers of 256 units and ReLU activations. The optimization is performed using Adam with a learning rate of 3×10^{-4} and a discount factor $\gamma = 0.99$.

4.2. Offline Dataset (Phase 3A)

For CQL, we generated a dataset of 50,000 transitions (200 episodes). To ensure diversity, we used a mixed behavior policy:

- 70% Random exploration.
- 30% Rule-based controller (heuristic battery management).

This mix is crucial for offline learning to cover both high-reward and low-reward regions of the state space.

4.3. Training

- **Offline:** CQL was trained for 500 epochs with a conservative weight $\alpha = 10.0$.
- **Online:** SAC and TD3 were trained for 350 episodes. We implemented strict action scaling using ‘tanh‘ to ensure outputs respect physical actuator limits.

5. Experiments and Results

We evaluated the trained agents over 20 test episodes. The comparison focuses on the trade-off between energy cost and thermal comfort.

5.1. Quantitative Analysis

Table 1 summarizes the performance. TD3 achieves the best (highest) reward, despite having the highest cost.

Table 1. Performance Comparison (Avg over 20 episodes)

Metric	CQL	SAC	TD3
Avg Reward	-447.37	-454.52	**-68.46**
Avg Cost ($)	**32.41**	32.88	62.32
Avg SOC	0.499	0.499	0.481
Comfort Violations	250.0	250.0	**0.0**

5.2. Reward vs. Cost Trade-off

As shown in Figure 1, TD3 consistently maintains a reward around -68, whereas SAC and CQL fluctuate significantly lower (worse) around -450.

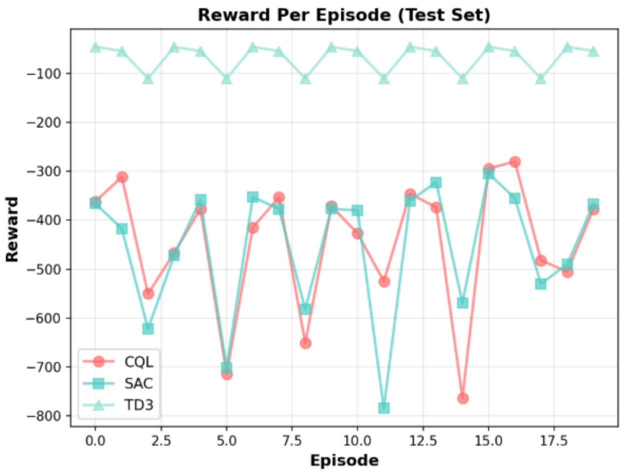

Figure 1. Reward per Episode on Test Set. TD3 (Top line) significantly outperforms SAC and CQL.

However, Figure 2 reveals an interesting inverse relationship. TD3 incurs significantly higher costs ($50-$90 per episode) compared to SAC and CQL ($20).

Analysis: The reward function includes a heavy penalty for discomfort (λ_c). SAC and CQL converged to a ”passive” policy: they turn off HVAC systems to minimize electricity cost (achieving $32), but this results in extreme temperatures, leading to massive comfort penalties. TD3, conversely, learned that paying for electricity to maintain comfort yields a better total reward.

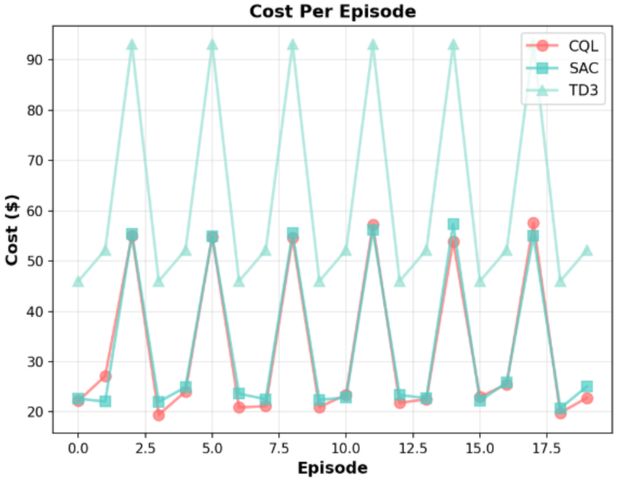

Figure 2. Cost per Episode. Note that the best performing agent in terms of reward (TD3) actually has the highest monetary cost.

5.3. Stability and Variability

Figure 3 illustrates the distribution of rewards. TD3 exhibits very low variance, indicating a stable, robust policy. CQL and SAC show high variance and poor median performance, indicative of failure to solve the control task effectively.

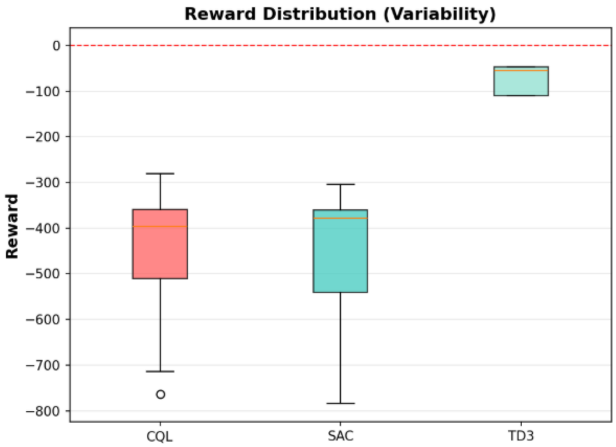

Figure 3. Reward Distribution Variability. TD3 shows high stability near 0, while others suffer massive penalties.

5.4. Battery Utilization

The battery usage analysis (Figure 4) further confirms the passive nature of SAC and CQL. Their average State of Charge (SOC) remains near the initialization point (0.5), represented by the dashed red line. This implies they are not utilizing the battery. TD3 shows a slight deviation (0.481), suggesting it is actively discharging the battery to offset some energy costs, though HVAC dominance masks this effect.

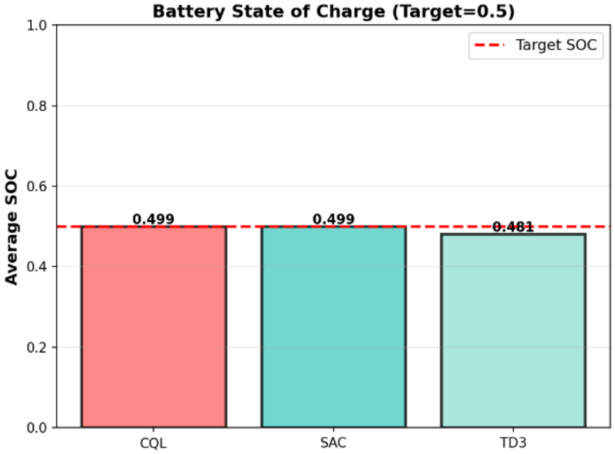

Figure 4. Average Battery State of Charge (SOC). SAC and CQL remain idle (0.5), while TD3 shows active usage.

6. Limitations

- **Offline Dataset Quality:** The failure of CQL is largely attributed to the dataset composition (70% random). Without sufficient examples of ”good” control, the conservative nature of CQL suppresses actions that might yield high comfort, defaulting to safe, low-energy actions.
- **Reward Shaping:** The drastic difference between TD3 and SAC suggests the reward landscape has deep local minima (zero cost) that entrap entropy-based exploration strategies.

7. Conclusion

We performed a comprehensive comparison of RL algorithms for Building Energy Management. Our results demonstrate that **TD3** is the most effective algorithm for this continuous control task, successfully prioritizing occupant comfort. We identified a critical failure mode in SAC and offline CQL, where agents minimize costs by sacrificing all comfort. Future work will focus on improving the offline dataset quality with expert demonstrations and refining reward weights to encourage balanced policies.

A. Appendix: Negative Results

A.1. The ”Do-Nothing” Convergence

A mandatory discussion in RL applications is algorithmic failure. In our experiments, SAC consistently converged to a policy that we describe as ”Do-Nothing.”

In the Sinergym environment, the electricity cost is strictly positive ($C_{elec} \geq 0$). The comfort penalty is non-negative ($P_{comfort} \geq 0$). The global maximum reward is 0.

SAC, which maximizes entropy, likely explored the state of ”HVAC OFF.” This results in $C_{elec} \approx 0$. While $P_{comfort}$ becomes high, the gradient to reduce $P_{comfort}$ requires increasing C_{elec} immediately. It appears SAC got stuck in the local optimum of minimizing C_{elec}, unable to bridge the gap to the region where high cost yields significantly lower penalties.

A.2. CQL Conservatism

CQL penalizes Q-values for actions not seen in the dataset. Since our dataset was 70% random (which often results in random costs and high discomfort), CQL effectively learned that ”doing something” is risky. Consequently, it learned to mimic the subset of data with the lowest variance, which corresponds to the passive, low-energy states, leading to poor thermal comfort performance.

References

Fujimoto, S., Hoof, H., and Meger, D. Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pp. 1587–1596. PMLR, 2018.

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pp. 1861–1870. PMLR, 2018.

Jiménez-Raboso, J., Campoy-Nieves, A., Manjavacas-Lucas, A., Gómez-Romero, J., and Molina-Solana, M. Sinergym: A building simulation and control interface for training reinforcement learning agents. *Proceedings of the 8th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, 2021.

Kumar, A., Zhou, A., Tucker, G., and Levine, S. Conservative q-learning for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1179–1191, 2020.

Mason, K. and Gdomb, S. Reinforcement learning with python. *Building Energy Management Systems*, 2019.