

Article

Model-Based Reinforcement Learning for HVAC Energy Optimization Under Hot, Mixed, and Cool Climates

Chengnan Lu  and Jinho Park * 

Department of Media, Soongsil University, Seoul 06978, Republic of Korea

* Correspondence: c2alpha@ssu.ac.kr; Tel.: +82-2-820-0935

Abstract

HVAC control trades energy against thermal comfort, complicated by two building features: thermal mass spreads a setpoint change over hours, and the input-to-outcome mapping shifts across the year. Model-free algorithms such as PPO, SAC, and TD3 carry no model of building dynamics and cannot evaluate a setpoint's downstream effect. We apply a Latent Dynamics Learning and Planning (LDLP) framework that learns a latent model of the building's thermal response and plans with Monte Carlo Tree Search inside it. We also gate a reward formulation common in prior work, applying its comfort penalty only while the building is occupied. On Sinergym's 5Zone environment under hot, mixed, and cool climates, LDLP is evaluated against PPO, SAC, TD3 and Sampled EfficientZero, an independent model-based controller run at the same budget. With energy normalized for the comfort achieved, LDLP consumes 4% to 26% less than PPO and SAC under the standard reward and 1% to 19% less under the gated reward. Running the same model with one simulation per decision, which removes planning, multiplies its normalized energy fivefold. Under the gated reward the deterministic TD3 policy degenerates onto a few fixed setpoints, so we report action diversity alongside the conventional metrics.

Keywords: HVAC energy optimization; model-based reinforcement learning; Monte Carlo Tree Search

1. Introduction

HVAC equipment is responsible for nearly half of the energy used in commercial and residential buildings [1]. Recent systematic reviews have catalogued the growing range of artificial-intelligence-enabled HVAC applications aimed at reducing this consumption toward zero-emission buildings [2]. Operating such systems efficiently requires balancing two objectives that trade off against each other: maintaining occupant thermal comfort requires vigorous HVAC response, while reducing energy consumption requires restraint. Neither side of the trade-off is static. Both shift continuously with outdoor temperature, humidity, solar radiation, and the daily and seasonal patterns of occupancy.

Concrete walls, slabs, and ceilings store substantial heat. They absorb solar and ambient gains during the day and release them slowly over the following hours. The indoor temperature therefore does not respond instantly to a setpoint change; depending on construction type, insulation, and outdoor conditions, the lag can extend from minutes to several hours, and stored heat radiating inward from structural mass can prolong the effect well into the evening. This delay is consequential for a reinforcement learning agent. When the impact of an action becomes visible only several timesteps after it was taken, credit assignment is unreliable unless the training data is both large and varied.



Academic Editor: Domenico Mazzeo

Received: 30 June 2026

Revised: 28 August 2026

Accepted: 13 September 2026

Published: 15 September 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons](https://creativecommons.org/licenses/by/4.0/)[Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Non-stationarity compounds the problem. Lowering the cooling setpoint on a summer afternoon imposes a heavy load because solar gain is at its peak, while the same input on a temperate spring morning produces minimal energy demand. A policy responding only to immediate reward cannot distinguish those situations without extensive experience across conditions, and even with that experience the input-to-outcome mapping it learns is statistical rather than causal.

Conventional approaches to this problem consist of rule-based controllers and model predictive control (MPC). Rule-based logic uses a fixed table of condition–action pairs; it is easy to deploy but cannot adapt beyond the cases its designers anticipated. MPC builds an explicit physical model of the building’s thermal dynamics, re-optimizes a finite-horizon trajectory at every step using weather forecasts, and then applies the first action of the optimal sequence. Empirical studies have reported energy savings over conventional reactive control that vary widely by building and baseline, commonly falling in the range of 20% to 30% but reaching as high as 50% to 60% in some cases [3]. The obstacle to wider adoption is scalability: every building requires its own model, calibrated to its specific geometry, materials, and equipment. Building and validating that model is engineering-intensive, and substantial recalibration is needed whenever the structure is renovated or its occupancy pattern shifts.

Deep reinforcement learning (DRL) provides an alternative approach in which a control policy is learned directly from simulator interaction, without requiring an explicit physical model [4,5]. Open benchmarking infrastructure such as Sinergym [6] and CityLearn [7] has made systematic evaluation of these methods practical, and Proximal Policy Optimization (PPO) [8], Soft Actor-Critic (SAC) [9], and Twin Delayed Deep Deterministic Policy Gradient (TD3) [10] have each been applied to HVAC control and reported to outperform rule-based baselines in such environments [11–13]. Despite these gains, model-free DRL has a structural limitation that is especially costly in HVAC control. With no internal model of how the environment evolves, the agent cannot anticipate how a given setpoint will alter the thermal state and energy demand over the hours that follow. Under a combined reward signal, the resulting policy is determined by whichever objective produces the stronger immediate response, and the balance the reward designer intended is not reliably enforced.

This paper applies a Latent Dynamics Learning and Planning (LDLP) framework to multi-zone HVAC control. The framework learns the building’s thermal dynamics in a compact latent state space and performs Monte Carlo Tree Search (MCTS) within that learned model. Candidate setpoints are evaluated for their downstream energy and comfort consequences before any action is executed in the real environment. Each rollout is conducted entirely within the learned model, which allows the agent to trace the propagation of a setpoint change through the building’s thermal mass over subsequent timesteps without invoking the simulator. We also modify a reward formulation common in prior HVAC reinforcement learning work whose comfort penalty remains active even when the building is unoccupied, gating that penalty by occupancy so that it applies only when occupants are present and no energy is spent maintaining the comfort band in an empty building. Experiments are conducted on the Sinergym 5Zone building model under hot, mixed, and cool climates, with PPO, SAC, and TD3 serving as model-free baselines and Sampled EfficientZero as an independently implemented model-based baseline.

Relative to the latent-model planning literature discussed in Section 2, this work makes two contributions. Planning inside a learned latent model has been developed on games and simulated robotics, where the predicted quantity is a game state or a rigid-body configuration; here the action is a continuous two-dimensional setpoint pair and the property that motivates planning is that an action’s effect is spread over hours by thermal

mass. The occupancy gate is a change to the objective rather than to the architecture, so it applies to any controller trained against it and is evaluated separately. These two interact: under the gated objective the baselines behave differently from one another, in a way that the conventional energy and comfort metrics do not reveal.

2. Related Work

2.1. Deep Reinforcement Learning for HVAC Control

Wei et al. [11] were among the first to apply deep reinforcement learning to HVAC control, formulating the problem as a Markov decision process (MDP) and addressing it with a Deep Q-Network (DQN) agent, regulating indoor airflow within EnergyPlus and achieving lower operational cost than a rule-based reference without constructing a physical model. Their setup, in which the problem is formulated as an MDP, an agent is trained in simulation, and its performance is compared against rule-based control, became the template adopted by much of the subsequent literature. DQN, however, operates on discrete action sets, which limits the resolution of the setpoints it can issue and effectively precludes continuous temperature regulation.

A more direct comparison among continuous-action algorithms was reported by Biemann et al. [12], who evaluated SAC, TD3, Trust Region Policy Optimization (TRPO), and PPO within a two-zone data center under varying weather. All four algorithms reduced energy consumption by at least 10% relative to the built-in model-based controller, though at some cost to temperature regulation: the baseline controller kept indoor temperatures within the required envelope more consistently than the RL policies did, illustrating the same energy-comfort trade-off this paper addresses. From a practical perspective, SAC attained equivalent performance using approximately one-tenth of the training data required by PPO. The underlying cause is the replay buffer: off-policy methods retain past transitions and reuse them across multiple gradient updates, whereas on-policy approaches discard each batch after a single update. Because building simulators are computationally expensive, such sample efficiency is critical. TD3, also off-policy, underperformed relative to SAC under stochastic conditions, which the authors interpreted as a limitation of deterministic policies in noisy environments. The study spanned several weather conditions, and a further experiment trained a single policy on weather drawn from roughly twenty cities in rotation to probe cross-region generalization; even so, no explicit model of the underlying dynamics was constructed, so generalization was pursued by exposing the policy to varied weather during training rather than by planning over a learned model.

Every method in this group learns a policy and nothing else, so the multi-step lag introduced by thermal mass is never represented explicitly and must be absorbed statistically into the policy. These studies also compare controllers on the same two aggregate quantities throughout, without any measure of whether the resulting controller varies its output with conditions.

2.2. Sinergym and Multi-Climate Benchmarking

Campoy-Nieves et al. [6] encapsulated EnergyPlus within a Gymnasium-compatible interface, incorporating observation and action normalization and stochastic weather perturbations, and released the result as Sinergym. Comprising multiple preconfigured environments that span various building types and climate files, the framework provides a common testbed for controlled algorithmic comparison.

Manjavacas et al. [13] benchmarked PPO, SAC, and TD3 on the 5Zone model across hot, mixed, and cool climates, broadly corroborating the findings of Biemann et al.: SAC outperformed the others in both reward and convergence speed, TD3 converged rapidly but plateaued below SAC, and PPO required more episodes to converge but settled into

a balanced trade-off between energy and comfort. Evaluating an agent under a climate other than its training climate degraded performance, though the authors characterized the resulting losses as comparatively minor and potentially acceptable in practice rather than a wholesale failure to generalize. A less anticipated finding was that progressive training across climates failed to yield improvement over single-climate training and in some cases degraded performance, with the resulting policy biased toward whichever climate had appeared most recently in the curriculum. Reversing the training order reversed the bias, confirming that earlier climate knowledge was being overwritten rather than accumulated. The authors suggested this pattern could reflect catastrophic forgetting, in which network weights drift toward the most recent training distribution at the expense of previously acquired knowledge. None of this work attempts to construct a latent model of the building's thermal response, nor does any plan over such a model before selecting an action.

Planning inside a learned latent model has its own line of work. MuZero learns the dynamics jointly with a value and a policy head and searches in the latent space [14]; EfficientZero adds a self-supervised consistency loss and value correction that make the same search work from far less data [15]; and Sampled MuZero extends the search to action spaces too large to enumerate by expanding a sampled subset of actions at each node [16]. Open implementations of these algorithms are collected in LightZero [17], which is the source of the Sampled EfficientZero baseline used here.

Beyond simulation-based benchmarks, deep reinforcement learning has also been deployed for real-time indoor temperature regulation using industrial communication protocols such as Modbus TCP, combining an LSTM temperature forecaster with a DQN controller [18]. Unlike this and the model-free approaches surveyed above, the framework introduced in this paper constructs a latent dynamics model and plans directly within it using Monte Carlo Tree Search.

3. Environment and Planning Framework Design

3.1. HVAC Control Environment

We adopt Sinergym's 5Zone building model as the experimental testbed, with EnergyPlus serving as the underlying physics engine. At every timestep, EnergyPlus integrates a system of coupled thermodynamic differential equations, computing zone-level air temperature, humidity, and HVAC power demand with high physical fidelity. The 5Zone model represents a single-story commercial building partitioned into east, west, south, north, and core zones, each characterized by its own facade exposure, glazing fraction, and occupancy schedule. The zones do not respond uniformly to identical outdoor weather, and thermal coupling through shared walls and air mixing implies that adjusting one zone's setpoint propagates into its neighbors. The control problem is therefore intrinsically multi-zone and cannot be decomposed into independent single-zone problems.

This building and these three climates are the configuration on which Manjavacas et al. benchmarked PPO, SAC and TD3, and the three baselines are the ones they and Biemann et al. used. Adopting the same testbed and the same comparison set keeps this study on a coordinate system an existing benchmark already occupies, so that differences in outcome can be attributed to the reward formulation and the control architecture rather than to a change of environment. The three baselines also span the design axes that matter here: PPO is on-policy while SAC and TD3 are off-policy, PPO and SAC maintain stochastic policies while TD3 is deterministic, and all three support continuous actions, which DQN-based approaches do not.

Each experiment is conducted with a single weather file (hot, mixed, or cool), and every algorithm is trained and evaluated independently under that climate. An episode spans a complete calendar year at a 15-min control resolution, comprising 35,039 timesteps. Within a single continuous training trajectory, every algorithm is therefore exposed to the summer cooling load, the shoulder seasons, and the winter heating load.

The observation vector comprises three categories of information. Temporal context is supplied as a six-dimensional vector of cosine–sine pairs representing month, day, and hour. This trigonometric encoding is necessary because the integer encoding of time is discontinuous at its endpoints: hour 23 and hour 0 are numerically far apart yet temporally adjacent, and a network cannot recover that continuity from raw integers. The weather data includes dry-bulb temperature, relative humidity, wind speed and direction, and diffuse and direct solar irradiance, the variables that govern envelope heat exchange and account for why identical setpoints yield different energy outcomes under different conditions. The building state contains per-zone cooling and heating setpoints, zone air and mean radiant temperatures, the Predicted Percentage Dissatisfied (PPD) comfort index, occupant count, and total HVAC power demand. PPD is computed from the Predicted Mean Vote (PMV) model of Fanger, standardized in ISO 7730 and reviewed alongside the other thermal comfort indices in [19], and provides a direct measure of occupant dissatisfaction that incorporates radiant temperature, humidity, and air velocity, rather than relying on simple temperature deviation. Comprehensive reviews of thermal-comfort-based building control strategies have found the PMV model to be the predominant modeling approach in the literature [20].

The action is a two-dimensional continuous vector containing a cooling setpoint in $[22.5, 30.0]$ °C and a heating setpoint in $[15.0, 22.5]$ °C, applied uniformly across every zone. Constraining the upper bound of heating to coincide with the lower bound of cooling prevents the system from heating and cooling simultaneously. The agent emits actions normalized to $[-1, 1]$, and the wrapper layer rescales them to the physical setpoint ranges.

The reward function adopted in much of the prior HVAC reinforcement learning literature combines an energy penalty and a comfort-violation penalty in a fixed weighted sum, as given by Equation (1):

$$r_t^{\text{std}} = -W \lambda_E E_t - (1 - W) \lambda_T T_t. \quad (1)$$

Here, E_t denotes the HVAC power demand in watts and T_t quantifies the deviation of the indoor temperature from the acceptable comfort band, defined as $[23, 26]$ °C in summer and $[20, 23.5]$ °C otherwise. The weight $W \in [0, 1]$ controls the relative importance of the two objectives and is set to 0.5 throughout. The constants $\lambda_E = 10^{-4}$ and $\lambda_T = 1.0$ bring the two penalties, expressed in different physical units, to comparable magnitude; without λ_E the energy term would exceed the comfort term by four orders of magnitude.

The formulation in Equation (1) has a limitation that has been largely overlooked. The comfort term remains active at every timestep regardless of whether the building is occupied. When the building is unoccupied, penalizing temperature deviation serves no practical purpose, as no occupant is present to experience discomfort; nevertheless, the policy is still driven to maintain conditions within the comfort band, thereby expending energy without any corresponding benefit. Under this reward, the agent has no incentive to widen the deadband during unoccupied hours, even though doing so aligns with the preference of a human operator.

We modify the reward such that the comfort term is gated by occupancy, as defined in Equation (2):

$$r_t = \begin{cases} -W \lambda_E E_t - (1 - W) \lambda_T T_t, & N_t > 0, \\ -W \lambda_E E_t, & N_t = 0, \end{cases} \quad (2)$$

where N_t denotes the number of occupants reported by the simulator at time t . When $N_t = 0$, the reward reduces to a pure energy penalty, permitting the agent to relax the indoor temperature toward the value that minimizes consumption. When $N_t > 0$, the original combined objective in Equation (1) is recovered, and the agent must trade energy against comfort as before. The occupancy signal is already present in the observation vector, and thus this revision requires no additional information. It removes a redundant component from the reward signal.

The rationale is that thermal comfort is defined by occupant experience: when no occupant is present there is no one for the comfort term to measure, so retaining it only diverts energy to holding a temperature band that no one is in. Equation (2) removes the term precisely in those intervals, leaving a pure energy objective from which the agent determines how far to relax the indoor temperature given the observed occupancy and thermal state.

3.2. Planning in a Learned Latent Model

MCTS [21–23] consists of a four-stage cycle, illustrated from left to right in Figure 1. Selection descends from the root, at each step selecting the child with the largest Upper Confidence Bound (UCB) score. UCB combines an estimated action value $Q(s, a)$ with an exploration bonus that increases as the visit count $N(s, a)$ decreases. Well-explored children are selected primarily on the basis of their value estimates, while seldom-visited children remain candidates owing to the exploration bonus. This prevents promising branches from being discarded prematurely. The red arrows in the figure trace the path of descent to the first leaf that has not yet been expanded.

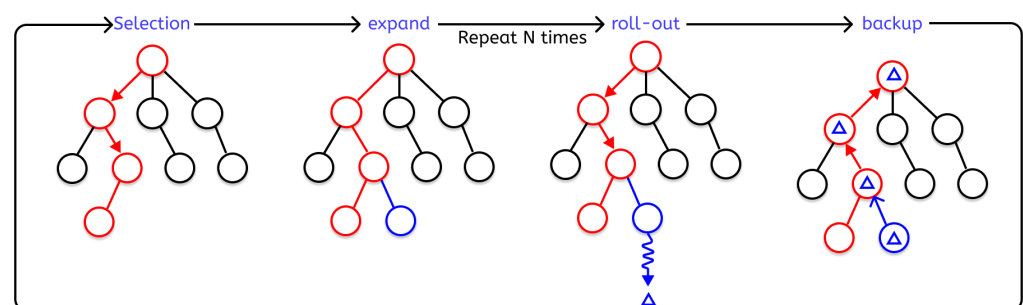


Figure 1. The four stages of MCTS, shown from left to right. During Selection, the algorithm follows the red arrows down the tree, selecting the child with the highest UCB score at each step. Expansion attaches the new blue child node beneath the selected leaf. Rollout, indicated by the blue wavy arrow, projects forward from that node to estimate its long-horizon value. Backup propagates the resulting estimate back up the selection path, with the blue triangles marking the nodes whose statistics are updated.

Expansion attaches a single new child, drawn in blue, representing an action not yet evaluated at the chosen leaf. Rollout then projects forward from that child to obtain an estimate of its long-horizon value; the blue wavy arrow and triangle in the figure depict this forward simulation. Backup propagates the rollout estimate backward along the path taken during Selection, incrementing visit counts and updating value estimates at every traversed node up to the root. The blue triangles ascending toward the root depict this propagation. Once a fixed number of simulations have been completed, the action with the

highest visit count at the root is committed to the environment. Visit counts are preferred over raw value estimates for action selection because they are less sensitive to the noisy early rollouts that occur before any branch has accumulated a sufficient number of visits.

Standard MCTS presupposes a fully known and deterministic transition function, as is the case in board games where each move has a precisely defined outcome [24]. A building environment violates this assumption in two respects. The true thermal dynamics are governed by differential equations that are not available in closed form outside the simulator itself, and invoking EnergyPlus at every candidate node during search would cause the rollout cost to grow linearly with both search depth and simulation count, precluding any meaningful lookahead across the 35,039 timesteps of an annual episode. We replace those environment calls with a learned latent dynamics model, following the design adopted in prior latent-model planning work [25,26], culminating in the three-component representation, dynamics, and prediction architecture illustrated in Figure 2. Efficient tensorized implementations of MCTS for such model-based agents, including batched GPU execution for MuZero-style search, have further reduced this computational overhead [27]. This latent-planning family has since been extended by sample-efficient successors such as EfficientZero. We adopt Sampled EfficientZero, which combines EfficientZero with the sampled expansion scheme for large action spaces, as the model-based baseline alongside the model-free controllers.

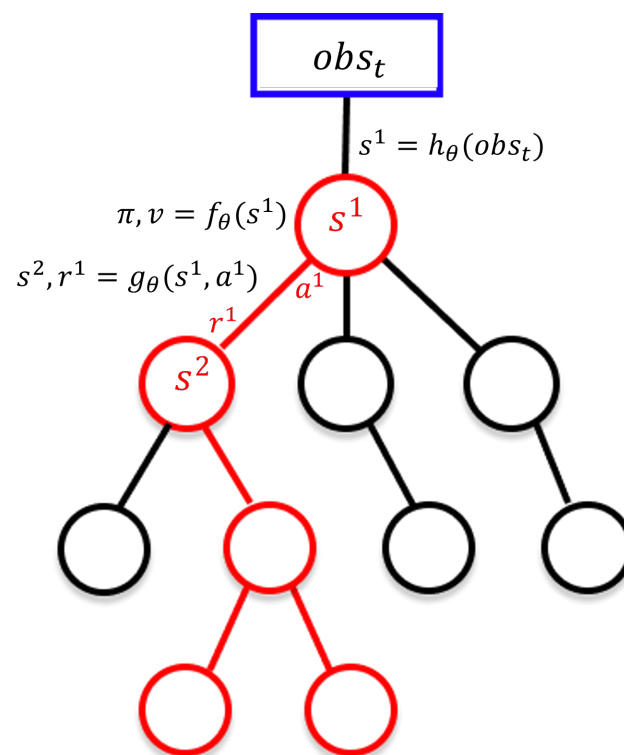


Figure 2. Latent dynamics model. The representation function maps the raw observation to a compact latent state. The dynamics function takes the current latent state and a candidate action and predicts the next latent state and the immediate reward, thereby replacing rollouts in the real environment. The prediction function produces a value estimate and a policy distribution at any latent state in the search tree. The blue rectangle denotes the raw observation, the red circles denote the latent states along one search path, and the black circles denote nodes not on that path.

The figure is interpreted from top to bottom. The blue rectangle at the top represents the raw observation. It is passed through the representation function h_θ , which compresses it into a compact latent state by removing noise and variables that do not contribute to the control decision. The first red circle below the rectangle corresponds to that latent

state. From this state, the dynamics function g_θ takes a candidate action and produces two outputs: the next latent state, which advances the building's thermal model forward in time, and the immediate reward, both predicted entirely in latent space without any call to EnergyPlus. Each subsequent red circle in the tree corresponds to a latent state obtained by repeated application of g_θ . The prediction function f_θ maps any latent state in the search tree to a long-horizon value estimate, which substitutes for the terminal rollout at leaf nodes, and to a policy distribution that biases the directions in which the tree is subsequently expanded. Because g_θ is fitted on real interaction data, it is in a position to capture the building's thermal lag structure: the multi-step delay between a setpoint change and its measurable effect on indoor temperature would be encoded in the learned weights themselves, allowing MCTS to trace the consequences of a setpoint several timesteps into the future without executing a real environment step. This constitutes the central difference from standard MCTS. The original algorithm cannot proceed without prior knowledge of subsequent states; g_θ removes this requirement by approximating the transition function from experience.

Continuous setpoint actions cannot be handled by the standard MCTS expansion scheme, which assumes a finite, enumerable set of children at each node. Progressive Widening [28] resolves this by tying the number of children a node may have to its visit count: a new child is added only once the visit count raised to the power α exceeds the current number of children. When visit counts are low, the tree stays narrow, so exploration concentrates on a few children and each accumulates enough visits to yield a stable value estimate; the tree widens only as this threshold is crossed repeatedly. The exponent α controls how fast this widening proceeds: smaller values keep the tree narrow for longer and devote the simulation budget to fewer, more thoroughly evaluated branches, whereas larger values expand it more aggressively at the cost of shallower per-node evaluation.

We use $\alpha = 0.4$, the default of the reference implementation on which our code is based rather than a value tuned for this problem. Its effect follows directly from that rule and depends on the simulation budget: applying the rule over the 100 simulations used for every reported result admits seven children at the root, so each control decision evaluates seven candidate setpoint pairs. Because the exponent governs only how the search spends its simulation budget, and not how the three networks are trained, its effect can be examined without retraining, which Section 5.4 does through the simulation budget.

Each new child action is sampled directly from the Gaussian policy distribution produced by the prediction function, following the sampled-action expansion scheme introduced for complex action spaces, so the tree grows toward the directions the current policy considers promising rather than uniformly across the action space. Figure 3 illustrates this sampling geometry for the two-dimensional HVAC setpoint space. The filled circle is the policy mean: the setpoint pair the network currently regards as most effective for the observed thermal state. Because high-probability regions of the distribution are sampled most often, new children fall predominantly near this mean. The surrounding ellipse outlines that high-probability region; setpoints beyond it are drawn only rarely, and enter the tree once enough visits have accumulated for Progressive Widening to reach them. Early in training, when policy uncertainty is high, the ellipse is broad and new children are scattered widely across the setpoint space. As training proceeds and the policy concentrates around its learned preferences, the ellipse contracts and new children cluster more tightly around the directions that have proven effective. Each sampled action is then passed to the action-normalization wrapper, which converts it into physical setpoints.

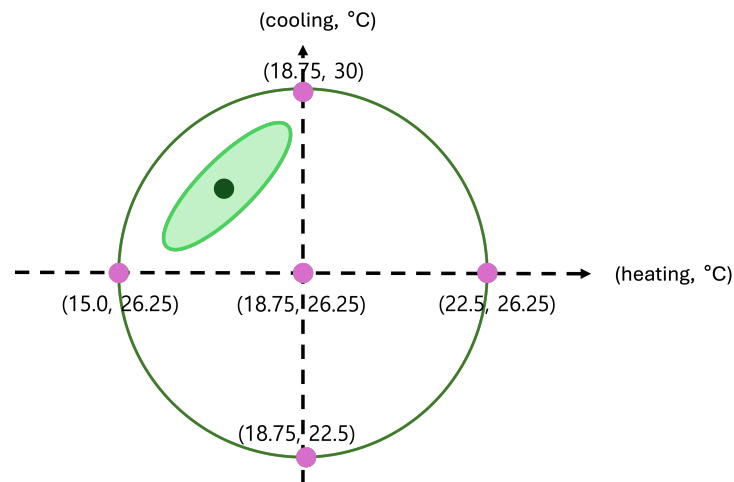


Figure 3. Action sampling from the two-dimensional Gaussian policy. The horizontal axis denotes the heating setpoint and the vertical axis the cooling setpoint. The filled circle indicates the distribution mean, the ellipse outlines the high-probability region from which child actions are most often sampled, and the green circle is a schematic boundary drawn for illustration only; the feasible action space is in fact the rectangle defined by the heating (15–22.5 °C) and cooling (22.5–30 °C) setpoint limits.

4. Latent Dynamics Learning and Planning Framework

The three networks are trained jointly under a single combined loss. The representation function maps observations to latent states through a multilayer perceptron with ELU activations, producing a 64-dimensional latent vector that is min–max rescaled to $[-1, 1]$ at every forward pass. The rescaling keeps latent magnitudes bounded across the many sequential applications of the dynamics function performed during search, which is what makes multi-step rollouts numerically stable. The representation function also learns to discard sensor noise and variables uninformative for control, yielding a representation compact enough for the dynamics function to predict its evolution accurately across multiple sequential rollout steps.

Given the current latent state and a candidate action, the dynamics function returns the next latent state and the immediate reward, without invoking EnergyPlus. The reward prediction head uses parameters separate from those of the state transition network. Predicting a scalar reward and predicting a high-dimensional latent transition are structurally distinct tasks whose optimal representations differ; sharing parameters between them induces gradient interference that degrades both predictions.

The prediction function produces a value estimate and a Gaussian policy distribution. The value estimate replaces the terminal rollout at leaf nodes, so a simulation can stop at a leaf rather than being rolled out to full depth. The policy distribution drives Progressive Widening by indicating which action directions are worth exploring next, with each new child node obtained as a fresh sample from this distribution. The log standard deviation of this distribution is clamped to a fixed range. If the variance collapses toward zero, the sampling distribution narrows until exploratory diversity is lost and the search converges prematurely to whatever the current policy already favors. If the variance instead grows too large, new child nodes are spawned faster than the simulation budget can evaluate them, producing a wide but shallow tree. Clamping confines the standard deviation to a range in which neither failure mode occurs. Table 1 collects the settings of the three networks described above together with the search and training parameters.

The combined loss comprises three terms. The reward term is the cross-entropy between the dynamics function’s predicted reward distribution and the categorical encoding of the reward observed in the environment; rewards and values are represented over a

discrete support rather than regressed as scalars, which is more robust to the wide dynamic range of the energy term. The value term is the corresponding cross-entropy against bootstrapped n -step returns, recomputed periodically with the current network so that stale targets in the replay buffer do not persist. The policy term drives the predicted Gaussian toward the distribution of actions selected at the root of the search, weighted by their visit counts, and an entropy term prevents premature narrowing of that Gaussian. All terms backpropagate through the shared representation function, so the latent space is organized around what planning requires rather than around what faithfully reconstructing the observations would require.

Table 1. LDLP hyperparameters. Values are those used for every reported result.

Parameter	Value	Parameter	Value
Latent dimension	64	MCTS simulations per decision	100
Representation MLP	[256, 128]	Progressive widening α	0.4
Dynamics MLP	[512, 512, 256, 128]	UCB c_1	1.25
Reward/value/policy heads	[128, 64] each	UCB c_2	19,652
Activation	ELU	Root exploration std scale	1.5
Observation stacking	4	Node prior	uniform
Optimizer	Adam	Policy log-std clamp	(−20, 0)
Weight decay	10^{-4}	Discount γ	0.99
Learning rate	10^{-3}	Unroll steps	4
LR decay	$\times 0.9/10,000$ steps	TD steps	4
Batch size	4096	Replay buffer	300 episodes
Training steps	30,000	Prioritized replay α	0.5
Self-play workers	4		

The three model-free baselines use the Stable-Baselines3 implementations, with the configurations given in Table 2.

Table 2. Baseline hyperparameters (Stable-Baselines3). The TD3 configuration differs from the other two in learning rate, discount factor and the absence of exploration noise; Section 5.5 examines the consequence.

Parameter	PPO	SAC	TD3
Learning rate	3×10^{-4}	3×10^{-4}	3×10^{-3}
Discount γ	0.99	0.99	0.9
Batch size	128	256	64
Network	[64, 64]	[256, 256]	[400, 300]
Replay buffer	—	10^6	10^6
Rollout length	4096	—	—
Epochs per update	10	—	—
GAE λ	0.95	—	—
Soft update τ	—	0.005	0.005
Action noise	—	—	none
Training	20 annual episodes (7.0×10^5 environment steps)		

5. Experimental Results

Each algorithm is trained and evaluated independently under each of the three climates, and under both reward formulations: the standard reward of Equation (1) in Table 3 and the occupancy-gated reward of Equation (2) in Table 4. A single policy per algorithm-climate-reward cell is trained and then evaluated over annual episodes of 35,039 timesteps each. Because Sinergym injects Ornstein–Uhlenbeck noise into the outdoor temperature series, the evaluation episodes differ from one another and the dispersion across them measures the sensitivity of a trained controller to weather variation. Alongside the three model-free baselines we evaluate Sampled EfficientZero as an independently implemented

model-based reference, taken from the LightZero library rather than written by us. It is given the same budget as LDLP—four parallel collectors, 100 simulations per decision, seven sampled actions at the root, batch size 4096 and 30,000 network updates—so the comparison is between two planning architectures at equal data and equal update counts, not between a tuned method and an untuned one.

Tables 3 and 4 report three quantities for every controller. Raw power consumption and comfort ratio each describe a different facet of behavior, but neither ranks the algorithms on its own. The comfort ratio is the fraction of occupied control steps ($N_t > 0$) at which the indoor temperature stays within the comfort band; unoccupied steps are excluded, consistent with the occupancy-gated reward. Power and comfort trade off against each other, so a low power figure may reflect efficient regulation or merely degraded comfort, and a high comfort ratio may reflect good control or merely unrestrained energy use. Energy is reported as the annual sum of the HVAC electricity demand E_t that Equation (1) penalizes, in units of 10^7 W summed over the control steps of a year. We therefore also report Power @ 100% Comfort, which divides that sum by the realized comfort ratio to express demand at a common comfort level, as defined in Equation (3):

$$\text{Power @ 100\% Comfort} = \frac{\sum_t E_t}{\text{Comfort Ratio}}. \quad (3)$$

This normalization is a first-order device rather than a principled efficiency measure: it assumes energy trades against comfort linearly and at a constant rate, and it over-corrects as the comfort ratio falls. It is therefore reported alongside the raw figures, and Section 5.3 repeats the comparison using dominance relations, which require no such assumption.

Table 3. Performance under the standard reward of Equation (1), the objective the model-free baselines were originally formulated against. Energy is the mean over the evaluation episodes with the sample standard deviation; the comfort ratio and the normalized figure are means over the same episodes. Power @ 100% Comfort is computed per episode by Equation (3) and then averaged, so it is not the quotient of the two preceding means: an episode with a low comfort ratio raises its own ratio and therefore the average. TD3 is set in grey because its policy degenerates (Section 5.2), so its figures are not directly comparable; bold marks the lowest energy and the lowest normalized energy among the remaining controllers.

Climate	Controller	Energy (10^7 W)	Comfort Ratio	Power @ 100% Comfort (10^7 W)
Hot	LDLP	8.408 ± 0.039	0.8017	10.489
	PPO	9.793 ± 0.046	0.7757	12.624
	SAC	8.832 ± 0.979	0.8795	12.165
	Sampled EfficientZero	7.034 ± 0.022	0.3154	22.305
	TD3	7.703 ± 0.046	0.7450	10.339
Mixed	LDLP	7.441 ± 0.105	0.7226	10.299
	PPO	10.454 ± 0.059	0.8676	12.050
	SAC	8.732 ± 0.958	0.8521	11.228
	Sampled EfficientZero	7.783 ± 0.062	0.3644	21.358
	TD3	4.456 ± 0.052	0.8303	5.367
Cool	LDLP	6.129 ± 0.054	0.7815	7.843
	PPO	9.039 ± 0.049	0.8477	10.663
	SAC	7.675 ± 0.104	0.9407	8.159
	Sampled EfficientZero	6.005 ± 0.052	0.3524	17.044
	TD3	10.757 ± 0.060	0.7526	14.293

Tables 3 and 4 report the same five controllers under the two objectives. Under the standard reward of Equation (1), the objective the model-free baselines were formulated against, LDLP records the lowest normalized figure of the controllers that keep modulating in all three climates. It reaches 10.489×10^7 W against 12.165 for SAC and 12.624 for PPO in

the hot climate, 10.299 against 11.228 and 12.050 in the mixed climate, and 7.843 against 8.159 and 10.663 in the cool climate. Its comfort ratio stays between 0.7226 and 0.8017, within 0.16 of SAC and above PPO in the hot climate.

Table 4. Performance under the occupancy-gated reward of Equation (2). Same controllers, climates and evaluation protocol as Table 3, with the same use of grey and bold.

Climate	Controller	Energy (10^7 W)	Comfort Ratio	Power @ 100% Comfort (10^7 W)
Hot	LDLP	6.607 $\pm$ 0.039	0.6604	10.005
	PPO	7.852 $\pm$ 0.044	0.7745	10.137
	SAC	6.978 $\pm$ 0.543	0.8568	10.349
	Sampled EfficientZero	7.041 $\pm$ 0.019	0.3169	22.225
	TD3	7.407 $\pm$ 0.038	0.7644	9.691
Mixed	LDLP	5.542 $\pm$ 0.096	0.6159	8.999
	PPO	6.124 $\pm$ 0.301	0.8063	10.087
	SAC	4.948 $\pm$ 0.114	0.7791	9.818
	Sampled EfficientZero	7.793 $\pm$ 0.070	0.3646	21.378
	TD3	4.363 $\pm$ 0.053	0.8579	5.086
Cool	LDLP	2.537 $\pm$ 0.043	0.7121	3.563
	PPO	3.542 $\pm$ 0.060	0.8590	4.123
	SAC	2.483 $\pm$ 0.071	0.8621	4.405
	Sampled EfficientZero	6.008 $\pm$ 0.054	0.3529	17.025
	TD3	2.576 $\pm$ 0.042	0.7369	3.496

Under the gated reward of Equation (2) the ordering is the same but the margins narrow. LDLP draws the least energy in the hot climate, 6.607×10^7 W, and among the controllers that keep modulating it again holds the lowest normalized figure in all three climates, 10.005, 8.999 and 3.563 against 10.137, 10.087 and 4.123 for PPO. The margin over PPO is 1.3% in the hot climate, its thinnest anywhere, and 10.8% and 13.6% in the mixed and cool climates. The gate also costs LDLP comfort: its ratio falls from 0.8017, 0.7226 and 0.7815 under Equation (1) to 0.6604, 0.6159 and 0.7121 here, which is the trade the gate instructs the controller to make when the building is empty.

The TD3 rows cannot be read as a direct comparison, which is why they are set in grey. TD3 records the lowest normalized figure in five of the six cells, and in the mixed climate under Equation (2) it records both the lowest energy and the highest comfort ratio of the five controllers. Neither outcome reflects better regulation. Section 5.2 shows the policy issues between 1 and 13 distinct setpoint pairs over an entire year under both objectives, with the four most frequent pairs covering at least 99.95% of the control steps, so those figures describe a controller that holds a fixed setpoint through every weather condition rather than one that trades energy against comfort. The remaining rows are therefore compared among themselves. The same caution applies to Sampled EfficientZero. Its comfort ratio stays between 0.3154 and 0.3646 in both tables, and at that comfort its raw energy is the lowest of all five controllers in the hot and cool climates under Equation (1), 7.034 and 6.005×10^7 W. Read on energy alone it would lead the table; read at a common comfort level it is last everywhere, between 17.025 and 22.305. This is precisely what the normalized column of Equation (3) attempts to expose, and it is the column on which the controllers should be compared.

5.1. Occupancy Behavior and the Distribution of Discomfort

Table 5 reports each quantity under both objectives, so every pair of columns isolates the effect of Equation (2). The two runs behind a pair share architecture, training budget and evaluation weather and differ only in the reward, which makes the difference between them attributable to the objective itself. The last pair carries the result the gate was designed

for. Unoccupied demand falls from 1284 to 714 W for LDLP in the hot climate, from 2111 to 1222 W in the mixed climate and from 2050 to 343 W in the cool climate, reductions of 44%, 42% and 83%. PPO falls by 46%, 57% and 76%, and SAC by 50%, 65% and 84%. The gate is therefore not exploited by LDLP alone: it is obeyed by every controller that still responds to the comfort term.

Table 5. Comfort held over occupied hours, and the split of power demand between occupied and unoccupied steps, under both rewards. Each quantity is given for Equations (1) and (2) side by side, so the two columns of a pair differ only in the objective the controller was trained against. Means over the evaluation episodes. TD3 is shown in grey for the reason given in Section 5.2.

Climate	Controller	Comfort Ratio		Excursion p95 (h)		Power Occupied (W)		Power Unoccupied (W)	
		Equation (1)	Equation (2)	Equation (1)	Equation (2)	Equation (1)	Equation (2)	Equation (1)	Equation (2)
Hot	LDLP	0.802	0.660	2.2	1.9	4164	3739	1284	714
	PPO	0.776	0.775	8.8	5.1	4233	4176	1885	1017
	SAC	0.879	0.857	2.1	3.0	3934	3862	1627	809
	Sampled EfficientZero	0.315	0.317	3.7	3.6	2917	2921	1432	1433
	TD3	0.745	0.764	0.3	13.9	4415	4085	797	868
Mixed	LDLP	0.723	0.616	2.1	2.8	2144	2151	2111	1222
	PPO	0.868	0.806	1.5	2.2	2627	2345	3209	1370
	SAC	0.852	0.779	2.8	3.2	2319	2189	2601	921
	Sampled EfficientZero	0.364	0.364	3.6	3.5	1838	1840	2463	2465
	TD3	0.830	0.858	2.2	1.6	2273	2233	638	621
Cool	LDLP	0.782	0.712	5.1	2.1	1274	1327	2050	343
	PPO	0.848	0.859	4.1	2.5	1326	1312	3372	821
	SAC	0.941	0.862	1.5	1.9	1245	1129	2788	443
	Sampled EfficientZero	0.352	0.353	3.9	3.9	891	890	2234	2237
	TD3	0.753	0.737	88.0	0.5	1565	1415	4022	305

The pair beside it shows what that saving did not cost. Occupied demand moves by at most 10.7% for the same three controllers in any climate, and LDLP's occupied demand is unchanged to within 4% in the mixed and cool climates, 2144 against 2151 W and 1274 against 1327 W. The two pairs read together are what makes the case for Equation (2): the energy it releases comes from hours in which no occupant is present to notice, and not from a controller that has simply widened its deadband, which would show the reduction in both pairs rather than in one.

Two controllers do not follow, and the paired comparison identifies them exactly. Sampled EfficientZero holds its comfort ratio and both power columns to within 0.5% between the two objectives, so the change in objective never reaches its policy at all; it also draws more power on the empty building than on the occupied one under both rewards in the mixed and cool climates, 2465 against 1840 W and 2237 against 890 W under Equation (2), which is the behavior the gate is meant to remove. TD3 moves in the wrong direction in the hot climate, where its unoccupied demand rises from 797 to 868 W, and barely moves in the mixed climate, from 638 to 621 W; the 92.4% reduction it records in the cool climate is obtained with the degenerate policy of Section 5.2, which repeats at most thirteen setpoint pairs over the year, so it is a property of a fixed setpoint rather than of a schedule. What the objective delivers therefore depends on the learner it is given to: the controllers that still respond to the comfort term follow it, and those that have stopped responding do not. Among the controllers that keep modulating, LDLP obtains the largest within-objective reduction in the cool climate, 74% against 37% for PPO and 61% for SAC, where unoccupied heating hours dominate the year.

The comfort side of the same table shows the price, and it is again clearest algorithm by algorithm. Moving from Equation (1) to Equation (2) costs LDLP 0.14, 0.11 and 0.07 of comfort ratio in the hot, mixed and cool climates, against just over 0.06 for PPO, whose ratio is unchanged in the hot climate and slightly higher in the cool one, and just under

0.08 for SAC. Apart from Sampled EfficientZero, whose ratio never reaches 0.37 under either objective, LDLP holds the lowest ratio of the five controllers in every climate under Equation (2). The discomfort it adds is not concentrated, however, and in two climates the gate shortens the spells rather than lengthening them: the 95th percentile of a continuous out-of-band spell falls from 2.2 to 1.9 hours in the hot climate and from 5.1 to 2.1 hours in the cool climate, rising only in the mixed climate, from 2.1 to 2.8 hours. Under Equation (2) those spells are shorter than PPO's 5.1 hours in the hot climate and far shorter than TD3's 13.9 hours there. The discomfort is spread thin rather than concentrated into long spells.

5.2. Action Distributions and the Retention of TD3

Aggregate metrics do not reveal how a controller regulates. Figures 4–6 plot the issued setpoint pairs as a scatter under Equation (2), one panel per controller, with the two model-based controllers on the top row and the three model-free ones below, and each point drawn at low opacity so that frequently visited setpoints accumulate into darker regions and stand in for a density. A setpoint visited a handful of times out of the 35,039 control steps of an evaluation episode does not read as sustained use; this is what lets the concentration of each policy be seen directly rather than inferred. Table 6 quantifies what they show.

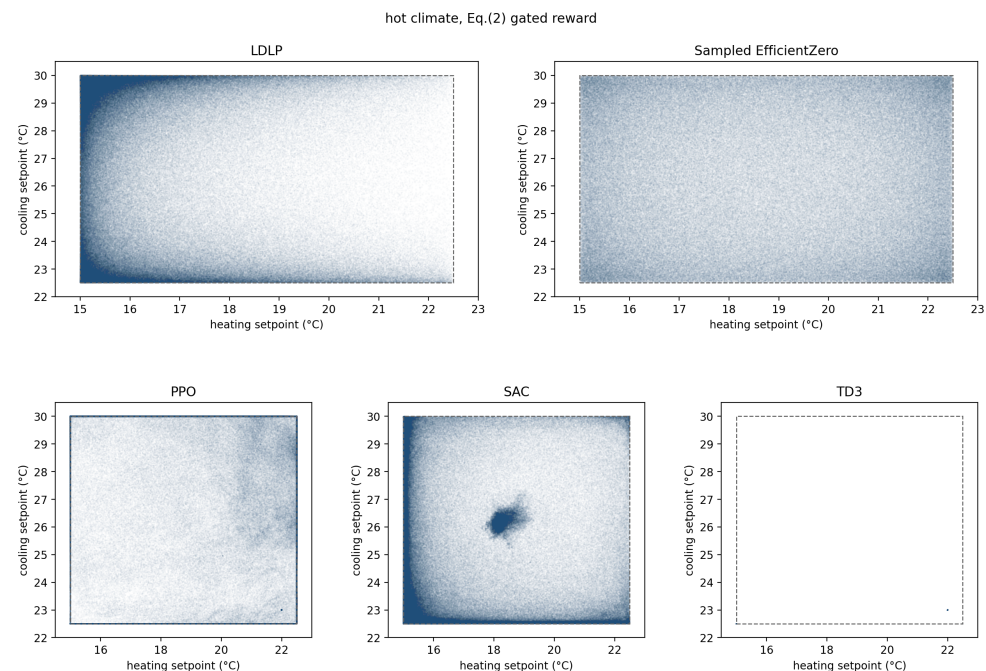


Figure 4. Issued setpoints under the hot climate and Equation (2), over the evaluation episodes, with the two model-based controllers on the top row and the three model-free ones below. Each point is one control step; points are drawn at low opacity, so setpoints visited often appear darker and stand in for a density, while the dashed rectangle marks the feasible heating–cooling setpoint box. In the TD3 panel the points collapse onto one or two setpoints and the rectangle is otherwise empty, whereas PPO, SAC, Sampled EfficientZero and LDLP fill it to differing extents.

TD3 issues 4, 13 and 1 distinct setpoint pairs per evaluation episode in the hot, mixed and cool climates, and its four most frequent pairs account for at least 99.95% of all control steps in every climate. Its entropy is 0.00 to 1.69 bits, against 9.71 to 11.06 for PPO and 14.44 to 14.47 for SAC. LDLP issues 18,456 to 30,713 pairs at 13.71 to 14.83 bits. Table 7 shows the same collapse under every retrained seed, so it is not an artifact of the single run reported here, and a controller whose output barely changes when the weather

changes is not tracking the weather. Its low energy is a consequence of that rather than of efficient regulation.

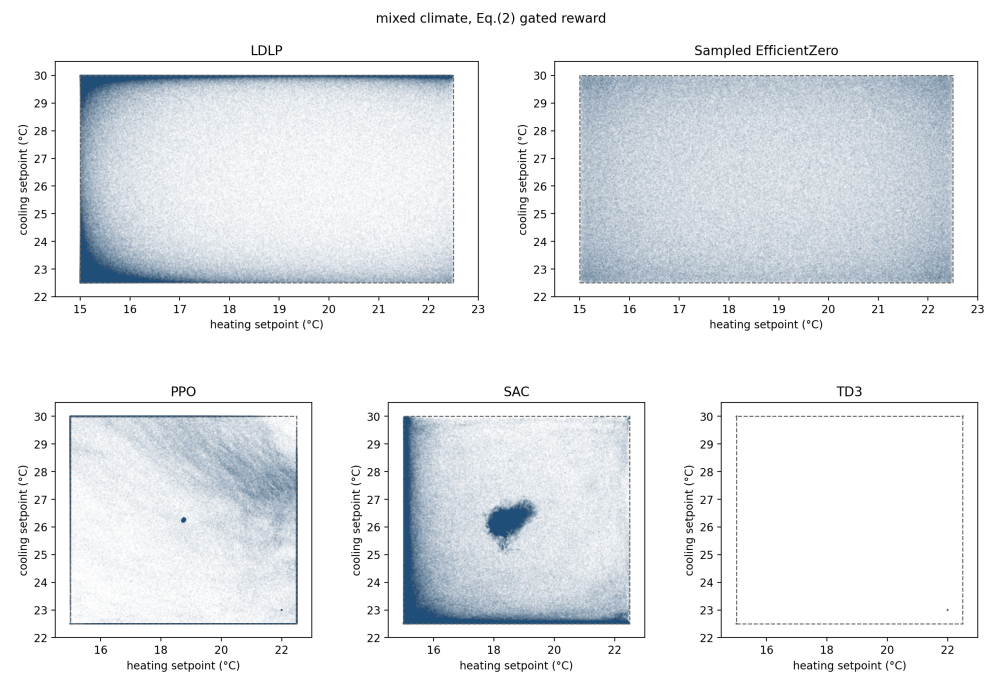


Figure 5. Issued setpoints under the mixed climate and Equation (2), over the evaluation episodes, in the layout of Figure 4.

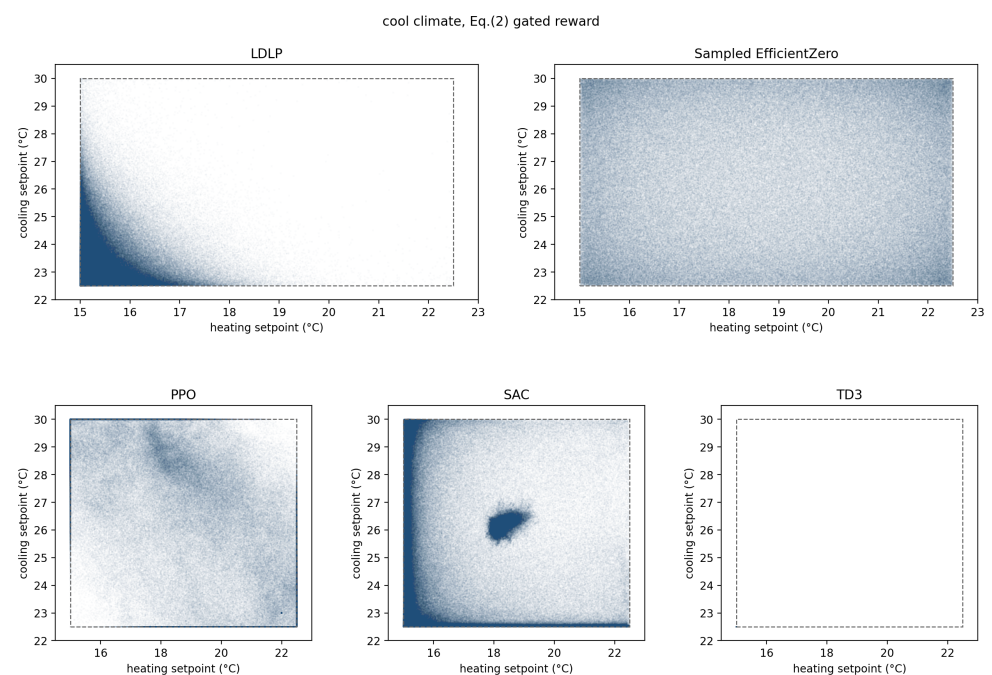


Figure 6. Issued setpoints under the cool climate and Equation (2), over the evaluation episodes, in the layout of Figure 4.

LDLP also places substantial mass near the boundary of the action rectangle, visible in Figures 4–6 as a darker band along the low-heating and high-cooling edges, which is expected: under Equation (2) the widest available deadband is optimal during the unoccupied hours that make up most of the year. What separates the two is diversity rather than concentration: LDLP keeps varying its selection with thermal conditions, three to four

orders of magnitude more finely than TD3. Diversity is necessary for that reading but not sufficient on its own, and Sampled EfficientZero is the case that shows it. Its entropy is the highest in Table 6, 15.03 bits in every climate, because it spreads setpoints almost uniformly over the rectangle, yet its normalized energy is the worst in Tables 3 and 4. A controller that ignores conditions can do so by repeating one setpoint or by scattering them at random; the diversity measure separates the first from a regulating policy, and the energy and comfort columns separate the second.

Table 6. Action-space occupancy under the occupancy-gated reward, per evaluation episode and averaged over them. Setpoint pairs are counted after rounding each setpoint to two decimals. TD3 is shown in grey.

Climate	Algorithm	Distinct Setpoints	Top-4 Share	Entropy (Bits)
Hot	LDLP	30,713	0.0008	14.83
	PPO	11,548	0.2000	11.06
	SAC	26,976	0.0027	14.47
	Sampled EfficientZero	33,871	0.0004	15.03
	TD3	4	1.0000	1.00
Mixed	LDLP	26,930	0.0023	14.49
	PPO	10,800	0.2846	9.71
	SAC	26,150	0.0013	14.44
	Sampled EfficientZero	33,860	0.0004	15.03
	TD3	13	0.9997	1.69
Cool	LDLP	18,456	0.0031	13.71
	PPO	15,073	0.2986	10.43
	SAC	26,405	0.0012	14.47
	Sampled EfficientZero	33,871	0.0004	15.03
	TD3	1	1.0000	0.00

Table 7. TD3 retrained from five independent seeds in every climate and under both rewards. Each cell gives the mean number of distinct setpoint pairs over the evaluation episodes, with the share held by the four most frequent pairs in parentheses. A share near one means the policy repeats a handful of setpoints for the whole year.

Reward	Climate	Seed 0	Seed 26	Seed 32	Seed 42	Seed 1000
Equation (1)	Hot	5986.4 (0.542)	35.6 (0.999)	7.0 (1.000)	1.0 (1.000)	1.0 (1.000)
	Mixed	68.6 (0.997)	162.2 (0.992)	14,312.8 (0.112)	3.1 (1.000)	63.6 (0.998)
	Cool	1.0 (1.000)	75.0 (0.997)	496.0 (0.970)	1.0 (1.000)	89.6 (0.996)
Equation (2)	Hot	59.8 (0.998)	1.0 (1.000)	1.0 (1.000)	4.2 (1.000)	1.0 (1.000)
	Mixed	24.0 (0.999)	35.4 (0.999)	1.0 (1.000)	12.6 (1.000)	1.0 (1.000)
	Cool	1.0 (1.000)	1.0 (1.000)	1.0 (1.000)	1.0 (1.000)	1.0 (1.000)

5.3. Pareto Analysis

A controller dominates another when it is at least as good on both axes and better on at least one: it uses no more energy and delivers no less comfort. Nothing has to be assumed about how the two are traded off. Any reader who values energy and comfort differently from us still reaches the same verdict, which is why dominance says more than a single normalized number. A controller that nothing dominates is called non-dominated.

As Table 8 shows, the answer depends on both the objective and the climate. Under the standard reward the front is wide. In the hot climate four of the five controllers are non-dominated and only PPO is not; in the mixed climate PPO, SAC and TD3 hold the front and LDLP is dominated by TD3; in the cool climate LDLP, SAC and Sampled EfficientZero hold it. Under the gated reward the front narrows. In the hot climate nothing dominates LDLP and nothing dominates SAC: LDLP uses less energy, 6.607 against 6.978×10^7 W, while SAC

holds the higher comfort ratio, so neither is better on both axes. In the mixed climate LDLP is dominated by SAC and by TD3, and in the cool climate by SAC, because those controllers match or beat it on energy while holding a higher comfort ratio. Sampled EfficientZero moves in the opposite direction between the two objectives. It is non-dominated in the hot and cool climates under Equation (1), where its raw energy is the lowest of the five, and dominated in every climate under Equation (2), where the other controllers cut their own demand and overtake it. That reversal is the limit of dominance as a criterion: it registers that no controller beats Sampled EfficientZero on both axes at once, but not that the comfort ratio it defends is only 0.32 to 0.36. The normalized column of Tables 3 and 4 is what prices that gap, and there Sampled EfficientZero is last in every cell.

The TD3 rows are the ones to read carefully, and they are why those rows are set in grey. Under the gated reward nothing dominates the controller of Section 5.2 in the mixed climate, though it repeats a handful of setpoint pairs for almost the whole year, and only SAC dominates it in the other two climates. Under the standard reward it is non-dominated in the hot and mixed climates as well. A controller that stops responding to the weather can therefore look as good on these two axes as one that regulates. Something that measures whether the controller reacts to conditions at all is needed alongside them, and that is the action diversity of Table 6.

Table 8. Pareto status of the learned controllers in the (energy, comfort) plane under both rewards, from the figures of Tables 3 and 4. Energy is minimized and comfort ratio maximized. TD3 is shown in grey for the reason given in Section 5.2.

Reward	Climate	Algorithm	Energy (10^7 W)	Comfort Ratio	Status
Equation (1)	Hot	LDLP	8.408	0.8017	non-dominated
		PPO	9.793	0.7757	dominated by LDLP, SAC
		SAC	8.832	0.8795	non-dominated
		Sampled EfficientZero	7.034	0.3154	non-dominated
		TD3	7.703	0.7450	non-dominated
	Mixed	LDLP	7.441	0.7226	dominated by TD3
		PPO	10.454	0.8676	non-dominated
		SAC	8.732	0.8521	non-dominated
		Sampled EfficientZero	7.783	0.3644	dominated by LDLP, TD3
		TD3	4.456	0.8303	non-dominated
	Cool	LDLP	6.129	0.7815	non-dominated
		PPO	9.039	0.8477	dominated by SAC
		SAC	7.675	0.9407	non-dominated
		Sampled EfficientZero	6.005	0.3524	non-dominated
		TD3	10.757	0.7526	dominated by LDLP, PPO, SAC
Equation (2)	Hot	LDLP	6.607	0.6604	non-dominated
		PPO	7.852	0.7745	dominated by SAC
		SAC	6.978	0.8568	non-dominated
		Sampled EfficientZero	7.041	0.3169	dominated by LDLP, SAC
		TD3	7.407	0.7644	dominated by SAC
	Mixed	LDLP	5.542	0.6159	dominated by SAC, TD3
		PPO	6.124	0.8063	dominated by TD3
		SAC	4.948	0.7791	dominated by TD3
		Sampled EfficientZero	7.793	0.3646	dominated by LDLP, PPO, SAC, TD3
		TD3	4.363	0.8579	non-dominated
	Cool	LDLP	2.537	0.7121	dominated by SAC
		PPO	3.542	0.8590	dominated by SAC
		SAC	2.483	0.8621	non-dominated
		Sampled EfficientZero	6.008	0.3529	dominated by LDLP, PPO, SAC, TD3
		TD3	2.576	0.7369	dominated by SAC

5.4. Ablation: Planning Budget and Widening

The two preceding sections read the controllers from the outside. This one removes parts of LDLP's own search and measures what each part contributes. The ablation reuses the trained model without any retraining; only the search is changed at decision time, so any difference is attributable to the search and not to a different set of weights.

The first ablation varies the number of MCTS simulations per decision. At one simulation the tree is never grown and the controller acts on the policy network alone, which is the framework without planning; at 25, 50 and 100 the search has a budget to spend. The same sweep is also an ablation of progressive widening, because the two are not independent: the rule adds a child only once the visit count reaches $(n + 1)^\alpha$, so the budget decides how many candidate actions the root ever holds. At one simulation the rule never fires and the root keeps a single candidate, which is the search running without widening; at 25, 50 and 100 it admits four, five and seven. Table 9 reports the mixed climate under the gated reward.

Removing the search costs most of the controller's performance. Without planning, the comfort ratio falls to 0.1504 and the normalized energy rises to 47.428×10^7 W, worse than any controller in Table 4, including Sampled EfficientZero at 21.378. Twenty-five simulations recover most of the gap, at 0.5525 and 10.008, and the remaining budget buys steady but smaller gains, ending at 0.6142 and 9.094 with 100 simulations. The energy drawn barely moves across the sweep, between 5.513 and 7.132×10^7 W; what the search buys is comfort at the same energy. Read as a widening ablation, the same rows say that the gain arrives with the first few candidates and then flattens: going from one candidate to four moves the comfort ratio from 0.1504 to 0.5525 and the normalized energy from 47.428 to 10.008×10^7 W, while going from four to seven moves them only to 0.6142 and 9.094. The action counts show that this is not a matter of the policy exploring more widely, since the number of distinct setpoints falls slightly as the budget grows: the search selects better among the candidates rather than producing more of them.

Table 9. Ablation of the planning budget and of progressive widening. The same trained LDLP model is evaluated with only the number of MCTS simulations per decision changed, in the mixed climate under Equation (2). The budget also fixes how far the widening rule of Section 3.2 can grow the root: applying $|C| < (n + 1)^\alpha$ over the budget admits 1, 4, 5 and 7 candidate actions at 1, 25, 50 and 100 simulations, so the first row is the model acting without planning and without any widening at all. The 100-simulation row is the default configuration re-evaluated for this sweep; it differs slightly from the corresponding row of Table 4 because the two draw different stochastic weather realizations. Bold marks the best value in each column across the sweep, both of which occur at 100 simulations.

Simulations	Energy (10^7 W)	Comfort Ratio	Power @ 100% Comfort (10^7 W)	Distinct Setpoints	Entropy (Bits)
1	7.132	0.1504	47.428	30,989	14.85
25	5.530	0.5525	10.008	27,337	14.53
50	5.513	0.5873	9.387	27,077	14.51
100	5.584	0.6142	9.094	26,942	14.49

The ablation therefore points to a single conclusion: what the controller gains comes from evaluating several candidate setpoints per decision rather than from the architecture that surrounds the search.

5.5. Reward Ablation and the Reproducibility of the TD3 Result

To separate the contribution of the occupancy gate from that of the planning architecture, every controller was trained under both reward formulations with nothing else changed. The two rewards share a single implementation and differ in one conditional, so any difference in behavior is attributable to the gate alone. Table 10 places each controller

side by side under the two objectives on energy, comfort and the number of distinct setpoint pairs; the reward values themselves are on different scales and are not comparable across the two.

The gate does what it was designed to do, and it does so for every controller that still modulates. LDLP draws 1.80, 1.90 and 3.59×10^7 W less under the gated objective in the hot, mixed and cool climates, PPO 1.94, 4.33 and 5.50, and SAC 1.85, 3.78 and 5.19. The comfort ratio falls with it, by 0.14 for LDLP in the hot climate and by less than 0.09 for PPO and SAC everywhere, which is the exchange the gate instructs the controller to make while the building is empty. Sampled EfficientZero is the exception in the other direction. Its energy and comfort barely move between the two objectives, 7.034 against 7.041 in the hot climate and 0.3154 against 0.3169, so the gate reaches a policy that is already insensitive to the comfort term.

The action counts isolate TD3. LDLP, PPO, SAC and Sampled EfficientZero issue between 8311 and 33,871 distinct setpoint pairs under both objectives, while TD3 issues between 1 and 13 under both. Its collapse is therefore not created by the gate. What the gate changes is how reliably the collapse appears, which the seed sweep below quantifies.

Table 10. Reward ablation for every controller. Each pair of columns gives the same quantity under the standard reward of Equation (1) and the occupancy-gated reward of Equation (2); the two arms differ only in whether the comfort term is gated. Energy is in 10^7 W. TD3 is shown in grey for the reason given in Section 5.2.

Climate	Algorithm	Energy		Comfort Ratio		Distinct Setpoints	
		Equation (1)	Equation (2)	Equation (1)	Equation (2)	Equation (1)	Equation (2)
Hot	LDLP	8.408	6.607	0.8017	0.6604	25,204	30,713
	PPO	9.793	7.852	0.7757	0.7745	8311	11,548
	SAC	8.832	6.978	0.8795	0.8568	28,091	26,976
	Sampled EfficientZero	7.034	7.041	0.3154	0.3169	33,854	33,871
	TD3	7.703	7.407	0.7450	0.7644	1	4
Mixed	LDLP	7.441	5.542	0.7226	0.6159	32,417	26,930
	PPO	10.454	6.124	0.8676	0.8063	10,583	10,800
	SAC	8.732	4.948	0.8521	0.7791	30,441	26,150
	Sampled EfficientZero	7.783	7.793	0.3644	0.3646	33,865	33,860
	TD3	4.456	4.363	0.8303	0.8579	3	13
Cool	LDLP	6.129	2.537	0.7815	0.7121	27,107	18,456
	PPO	9.039	3.542	0.8477	0.8590	10,295	15,073
	SAC	7.675	2.483	0.9407	0.8621	29,231	26,405
	Sampled EfficientZero	6.005	6.008	0.3524	0.3529	33,867	33,871
	TD3	10.757	2.576	0.7526	0.7369	1	1

Two questions attach to this result: the evaluation episodes vary the weather but not the training seed, and the TD3 configuration in Table 2 is not symmetric with the other two. Retraining TD3 from five independent seeds in every climate confirms that the degeneration is not an artifact of a single run. Under the gated objective all fifteen seed-climate cells degenerate. The policy issues between 1 and 60 distinct setpoint pairs, its four most frequent pairs cover at least 99.8% of the year, and all five seeds collapse onto a single pair in the cool climate. Under the standard reward thirteen of the fifteen cells behave the same way, the two exceptions issuing 5986 and 14,313 pairs, so the gate makes the collapse consistent rather than causing it outright. Aligning TD3's learning rate, discount and exploration noise with the other baselines removes the degeneration entirely, so it requires both the gated objective and a deterministic policy trained without exploration noise. A better-configured TD3 would therefore be a stronger comfort baseline

than the one reported in Table 4; we flag this rather than replace the rows the rest of the analysis rests on.

Table 7 gives the seed sweep in full. Under the gated objective all fifteen seed-climate cells degenerate, and in the cool climate every seed collapses onto one setpoint pair for the entire year. Under the standard reward thirteen of the fifteen cells behave the same way, the exceptions being the hot climate at seed 0 and the mixed climate at seed 32, which retain 5986 and 14,313 pairs. The gate therefore does not create the failure, it removes the cases in which the configuration escapes it.

6. Discussion

An agent's behavior is most reliably assessed by examining its action distribution together with the aggregate energy and comfort metrics. Power consumption summarizes how much energy a policy uses, while the scatter and density plots reveal how it regulates the building. The three model-free baselines occupy distinct regions of the setpoint space. SAC spreads its decisions broadly across the action rectangle, PPO concentrates along the upper and lower edges, and TD3 collapses onto a small set of corner setpoints. The two model-based panels differ again: LDLP concentrates along the low-heating and high-cooling edges that the gate rewards while still covering the interior, whereas Sampled EfficientZero fills the rectangle almost uniformly, which is breadth without selection. Different as these patterns are, they share a common cause. A model-free training signal arrives from the reward observed immediately after the most recent decision. The policy is therefore shaped chiefly by near-term feedback and does not represent the building's thermal lag, under which a setpoint change continues to affect indoor temperature and energy demand for several timesteps. LDLP differs in exactly this respect: each setpoint is selected by planning forward in the learned latent model, so the delayed consequences of a decision are evaluated before it is committed. Section 5.4 supports this attribution directly, since removing nothing but the planning step costs most of the comfort performance.

The margin is thinnest in the hot climate under the gated reward, where LDLP leads PPO by 1.3% on the normalized measure (Table 4). The mixed climate is weak in a different sense: under the same reward the Pareto analysis of Table 8 shows LDLP dominated there by SAC and by TD3. More generally, planning should be expected to pay off only where its premises hold. In a lightweight building the lag it exploits is short. Where occupancy is nearly constant, the gate of Equation (2) reduces to Equation (1). Where the control interval is short, the simulation budget must shrink until the search is too shallow to see past the immediate step.

Relative to the closest published comparison the ranking changes. Manjavacas et al. evaluated the same three baselines on the same building and climates under Equation (1) and reported SAC ahead with TD3 plateauing; under the gated reward we observe TD3 moving past a plateau into degeneration. As Section 5.5 shows, this degeneration is not a property of the gated reward alone. It arises from the combination of that reward with the particular TD3 configuration adopted here, a deterministic policy trained without exploration noise, and aligning that configuration with the other baselines removes it entirely. The reported TD3 rows should therefore be read as one specific baseline rather than as the best attainable TD3. The two studies are not comparable in absolute numbers since the objective differs, but the change in qualitative outcome is itself the point. Two published figures give a sense of scale. Campoy-Nieves et al. reported 15–18% savings for DRL over reactive control on a data center model, and the MPC literature surveyed by Kim et al. reports 20–30% as typical. Our normalized reductions against PPO reach 26% in the cool climate under the standard reward. The buildings, objectives and baselines differ

too much for a like-for-like comparison, so these numbers place the result on a scale rather than rank it.

The comparison against an independently implemented model-based method is carried by Sampled EfficientZero, which is taken from LightZero and run at the same data and update budget as LDLP. It does not transfer to this problem. Its comfort ratio stays between 0.3154 and 0.3646 under both rewards. Its normalized energy is the worst of the five controllers in every cell of Tables 3 and 4, and it is dominated in every climate under the gated reward in Table 8. Table 5 adds that its comfort ratio and its occupied and unoccupied demand all move by less than 0.5% when the objective changes. Two differences plausibly account for this. Its search samples K actions from the policy at the root and never adds more, whereas progressive widening lets the root grow candidates as visits accumulate. Its latent state is also trained without the reward-prediction weighting used here. We therefore report it as evidence that a model-based architecture is not by itself sufficient on this problem. The planning-budget ablation of Section 5.4 complements that reading, since it holds architecture, computation and tuning quality constant by construction.

Several limitations bound these findings. The evaluation is conducted entirely in simulation, and each of the conditions that distinguish a real installation bears on a specific component. Actuator delay and deadband hysteresis would misalign the response g_θ learned to predict, biasing every rollout within the planning horizon. Sensor noise, drift and dropout shift the input distribution of h_θ , and a latent state outside the region seen during training invalidates the search built on it. Structural mismatch between the EnergyPlus model and the building it represents means the search optimizes consequences that will not occur. Occupant disturbances such as opened windows break the correspondence between the scheduled occupancy signal and actual conditions, which the gate depends on. And 100 simulations per decision may exceed the budget of an embedded building-management controller. The occupancy gate has a structural limitation of its own: the signal N_t is read from a single zone while setpoints are applied to all five, so “the building is empty” is an approximation. Each climate is also treated as a separate problem. Whether one model can be trained to generalize across climates remains open, as does whether separating dynamics from policy mitigates the catastrophic forgetting reported by Manjavacas et al. Finally, the experiments use only the 5Zone building. The framework paper on which this testbed is based demonstrates its own experiments on a single building model, but the limitation is real. The remaining Sinergym models would each test a different aspect, and extending the framework to those building models is left as future work.

7. Conclusions

This paper applied a Latent Dynamics Learning and Planning framework to multi-zone HVAC control and evaluated it on Sinergym’s 5Zone building model under hot, mixed, and cool climates. The baselines were PPO, SAC and TD3, together with Sampled EfficientZero as an independently implemented model-based reference. Every controller was trained under both the standard reward and an occupancy-gated reward that applies the comfort penalty only while the building is occupied. Under the standard reward LDLP recorded the lowest energy normalized for the comfort achieved of any controller that continued to modulate, in all three climates. It kept that position under the occupancy-gated reward in all three climates, though the margin fell to 1.3% over PPO in the hot climate. Its comfort ratio was the lowest of those controllers under the gated objective, apart from Sampled EfficientZero, whose ratio stays between 0.32 and 0.36 under both objectives. That lower ratio is the trade the gate instructs LDLP to make while the building is empty. A single simulation per decision removes planning while holding every other component fixed. Run that way, the same trained model lost 46 percentage points of comfort ratio in

the mixed climate, and its normalized energy rose from 9.094 to 47.428×10^7 W. That places the gain in the search rather than in the surrounding architecture.

A second finding concerns evaluation. Under the gated objective the deterministic TD3 policy degenerated onto 1 to 13 distinct setpoints, and yet it remained non-dominated on energy and comfort in the mixed climate. Reporting energy and comfort alone is therefore not sufficient to distinguish a controller that regulates a building from one that does not, and a measure of action diversity is needed alongside them.

These findings are established in simulation, on one building model, with a single trained policy per condition evaluated across stochastic weather, and should be read as evidence of feasibility rather than of deployment readiness. Whether the advantage survives hyperparameter alignment of the baselines, other building models, and the transition from simulation to a real installation are the questions that follow.

Author Contributions: Conceptualization, C.L. and J.P.; methodology, C.L.; software, C.L.; validation, C.L. and J.P.; formal analysis, C.L.; investigation, C.L.; writing—original draft preparation, C.L.; writing—review and editing, J.P.; visualization, C.L.; supervision, J.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the Graduate School of Metaverse Convergence support program (IITP-2026-RS-2024-00430997) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data supporting the reported results are contained within the article. The complete hyperparameter configurations are given in Tables 1 and 2, and the implementation, reward definitions, and per-timestep evaluation logs are available from the authors on reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Xu, S.; Fu, Y.; Wang, Y.; Yang, Z.; Huang, C.; O'Neill, Z.; Wang, Z.; Zhu, Q. Efficient and assured reinforcement learning-based building HVAC control with heterogeneous expert-guided training. *Sci. Rep.* **2025**, *15*, 7677. [\[CrossRef\]](#) [\[PubMed\]](#)
2. Gassar, A.A.A.; Jafar, R. Artificial intelligence-enabled heating, ventilation, and air conditioning systems toward zero-emission buildings: A systematic review of applications, challenges, and future directions. *Appl. Sci.* **2025**, *15*, 10497. [\[CrossRef\]](#)
3. Kim, D.; Lee, J.; Do, S.; Mago, P.J.; Lee, K.H.; Cho, H. Energy modeling and model predictive control for HVAC in buildings: A review of current research trends. *Energies* **2022**, *15*, 7231. [\[CrossRef\]](#)
4. Zhong, X.; Zhang, Z.; Zhang, R.; Zhang, C. End-to-end deep reinforcement learning control for HVAC systems in office buildings. *Designs* **2022**, *6*, 52. [\[CrossRef\]](#)
5. Wang, Z.; Hong, T. Reinforcement learning for building controls: The opportunities and challenges. *Appl. Energy* **2020**, *269*, 115036. [\[CrossRef\]](#)
6. Campoy-Nieves, A.; Manjavacas, A.; Jiménez-Raboso, J.; Molina-Solana, M.; Gómez-Romero, J. Sinergym: A virtual testbed for building energy optimization with reinforcement learning. *Energy Build.* **2025**, *327*, 115075. [\[CrossRef\]](#)
7. Vázquez-Canteli, J.R.; Dey, S.; Henze, G.; Nagy, Z. CityLearn: Standardizing research in multi-agent reinforcement learning for demand response and urban energy management. *arXiv* **2020**, arXiv:2012.10504.
8. Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; Klimov, O. Proximal policy optimization algorithms. *arXiv* **2017**, arXiv:1707.06347. [\[CrossRef\]](#)
9. Haarnoja, T.; Zhou, A.; Abbeel, P.; Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In Proceedings of the 35th International Conference on Machine Learning (ICML), Stockholm, Sweden, 10–15 July 2018; pp. 1861–1870.
10. Fujimoto, S.; van Hoof, H.; Meger, D. Addressing function approximation error in actor-critic methods. In Proceedings of the 35th International Conference on Machine Learning (ICML), Stockholm, Sweden, 10–15 July 2018; pp. 1587–1596.
11. Wei, T.; Wang, Y.; Zhu, Q. Deep reinforcement learning for building HVAC control. In Proceedings of the 54th Annual Design Automation Conference (DAC '17), Austin, TX, USA, 18–22 June 2017.

12. Biemann, M.; Scheller, F.; Liu, X.; Huang, L. Experimental evaluation of model-free reinforcement learning algorithms for continuous HVAC control. *Appl. Energy* **2021**, *298*, 117164. [[CrossRef](#)]
13. Manjavacas, A.; Campoy-Nieves, A.; Jiménez-Raboso, J.; Molina-Solana, M.; Gómez-Romero, J. An experimental evaluation of deep reinforcement learning algorithms for HVAC control. *Artif. Intell. Rev.* **2024**, *57*, 173. [[CrossRef](#)]
14. Schrittwieser, J.; Antonoglou, I.; Hubert, T.; Simonyan, K.; Sifre, L.; Schmitt, S.; Guez, A.; Lockhart, E.; Hassabis, D.; Graepel, T.; et al. Mastering Atari, Go, chess and shogi by planning with a learned model. *Nature* **2020**, *588*, 604–609. [[CrossRef](#)] [[PubMed](#)]
15. Ye, W.; Liu, S.; Kurutach, T.; Abbeel, P.; Gao, Y. Mastering Atari games with limited data. In Proceedings of the Advances in Neural Information Processing Systems 34 (NeurIPS), Virtual, 6–14 December 2021; pp. 25476–25488.
16. Hubert, T.; Schrittwieser, J.; Antonoglou, I.; Barekatin, M.; Schmitt, S.; Silver, D. Learning and planning in complex action spaces. In Proceedings of the 38th International Conference on Machine Learning (ICML), Virtual, 18–24 July 2021; pp. 4476–4486.
17. Niu, Y.; Pu, Y.; Yang, Z.; Li, X.; Zhou, T.; Ren, J.; Hu, S.; Li, H.; Liu, Y. LightZero: A unified benchmark for Monte Carlo tree search in general sequential decision scenarios. In Proceedings of the Advances in Neural Information Processing Systems 36 (NeurIPS), New Orleans, LA, USA, 10–16 December 2023.
18. Kim, H.; Ejaz, M.A.; Lee, K.; Cho, H.-M.; Kim, D.H. Predictive optimal control mechanism of indoor temperature using Modbus TCP and deep reinforcement learning. *Appl. Sci.* **2025**, *15*, 7248. [[CrossRef](#)]
19. Streinu-Cercel, A.; Costoiu, S.; Mârza, M.; Streinu-Cercel, A.; Mârza, M. Models for the indices of thermal comfort. *J. Med. Life* **2008**, *1*, 148–156. [[PubMed](#)]
20. Grassi, B.; Piana, E.A.; Lezzi, A.M.; Pilotelli, M. A review of recent literature on systems and methods for the control of thermal comfort in buildings. *Appl. Sci.* **2022**, *12*, 5473. [[CrossRef](#)]
21. Coulom, R. Efficient selectivity and backup operators in Monte-Carlo tree search. In *Proceedings of the Computers and Games (CG 2006)*, LNCS 4630; Springer: Berlin/Heidelberg, Germany, 2007; pp. 72–83.
22. Kocsis, L.; Szepesvári, C. Bandit based Monte-Carlo planning. In *Machine Learning: ECML 2006*, LNCS 4212; Springer: Berlin/Heidelberg, Germany, 2006; pp. 282–293.
23. Browne, C.B.; Powley, E.; Whitehouse, D.; Lucas, S.M.; Cowling, P.I.; Rohlfshagen, P.; Tavener, S.; Perez, D.; Samothrakis, S.; Colton, S. A survey of Monte Carlo tree search methods. *IEEE Trans. Comput. Intell. AI Games* **2012**, *4*, 1–43. [[CrossRef](#)]
24. Silver, D.; Hubert, T.; Schrittwieser, J.; Antonoglou, I.; Lai, M.; Guez, A.; Lanctot, M.; Sifre, L.; Kumaran, D.; Graepel, T.; et al. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science* **2018**, *362*, 1140–1144. [[CrossRef](#)] [[PubMed](#)]
25. Oh, J.; Singh, S.; Lee, H. Value prediction network. In Proceedings of the Advances in Neural Information Processing Systems 30 (NeurIPS), Long Beach, CA, USA, 4–9 December 2017; pp. 6118–6128.
26. Hafner, D.; Lillicrap, T.; Fischer, I.; Villegas, R.; Ha, D.; Lee, H.; Davidson, J. Learning latent dynamics for planning from pixels. In Proceedings of the 36th International Conference on Machine Learning (ICML), Long Beach, CA, USA, 9–15 June 2019; PMLR 97, pp. 2555–2565.
27. Baláž, M.; Tarábek, P. Tensor implementation of Monte-Carlo tree search for model-based reinforcement learning. *Appl. Sci.* **2023**, *13*, 1406. [[CrossRef](#)]
28. Couëtoux, A.; Hoock, J.-B.; Sokolovska, N.; Teytaud, O.; Bonnard, N. Continuous upper confidence trees. In *Learning and Intelligent Optimization (LION 5)*, LNCS 6683; Springer: Berlin/Heidelberg, Germany, 2011; pp. 433–445.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.