arXiv:2607.16534v1 [cs.LG] 17 Jul 2026

Building2Building: A Large Scale Benchmark for Generalizable Real-World Reinforcement Learning

Vincent Taboga[1,2], **Justin Veilleux** [1,2], **Doseok Jang**[1,2], **Anushree Rankawat**[2], **Pierre-Luc Bacon**[1,2]
vincent.taboga@polymtl.ca

[1]**Mila - Quebec AI Institute**
[2]**Université de Montréal - Département d'Informatique et de Recherche Opérationnelle**

Abstract

Reinforcement learning (RL) has achieved strong results in control, yet learned policies remain brittle to changes in dynamics, action spaces, observation spaces, or goals, a critical limitation for real-world deployment. Existing benchmarks offer limited diversity and complexity, making it difficult to rigorously study transfer, multi-task learning, and meta-learning in RL. We introduce Building2Building[1] (*B2B*), a large-scale suite of realistic Heating, Ventilation, and Air Conditioning (HVAC) control environments built on EnergyPlus, a state-of-the-art building simulator. *B2B* is fully compatible with the Gymnasium interface and features a parametric building generator, enabling the systematic generation of diverse building configurations with heterogeneous observation and action spaces. Based on this suite, we define benchmark tasks targeting key open challenges in RL, including goal adaptation, dynamics adaptation, action-space shifts, and cross-domain transfer. By providing a large-scale, diverse, and physically grounded testbed with standardized evaluation protocols, *B2B* enables systematic investigation of generalization and transfer in continuous control. Beyond advancing research on generalization in RL, this new benchmark also carries significant societal implications by enabling improved HVAC control at scale, one of the most energy-intensive systems in buildings.

1 Introduction

Reinforcement Learning (RL) has achieved strong performance across a wide range of control problems. In most cases, however, these results are obtained by training a policy on a single target task. While effective in controlled settings, this paradigm limits the ability of RL systems to operate in new environments or adapt to related tasks. Adaptation is a key feature for real-world deployments, and enabling RL agents to generalize and adapt across various environments remains a major open challenge.

Studying generalization in RL requires simulated environments that expose agents to many tasks with varying dynamics, observation spaces, action spaces, and objectives. Existing benchmarks such as Meta-World+ (McLean et al., 2025) and RLBench (James et al., 2020) provide useful testbeds, but remain largely confined to the robotics domain. Moreover, the variations they introduce are often limited to changes in dynamics parameters (e.g., friction) or task goals such as target positions for navigation or object placement. As a result, these benchmarks offer limited diversity in environment structure and rarely include heterogeneous observation and action spaces. Because of this limitation,

[1]Code available at https://github.com/vtaboga/building2building

many works studying cross-domain adaptation or transfer in RL rely on custom experimental setups. These environments are often designed for a specific method and are difficult to reproduce, which makes comparisons across approaches challenging. Consequently, progress toward generalizable RL agents remains fragmented.

In this work, we propose a new domain for studying generalization in RL: heating, ventilation, and air-conditioning (HVAC) control in buildings. Buildings exhibit large diversity in layouts, materials, climate exposure, HVAC configurations and objectives balancing energy savings and comfort. At the same time, they share common physical principles based on thermodynamics and energy conservation. This combination of diverse dynamics, heterogeneous control interfaces, and shared physical principles makes HVAC control an ideal domain for studying generalization.

We introduce Building2Building (*B2B*), a large-scale suite of RL environments built on EnergyPlus, a state-of-the-art building energy simulator (NREL, 2017). EnergyPlus provides high-fidelity physical modeling of building thermal behavior. The simulator's accuracy is highlighted by works such as Zhang et al. (2019), which showed that RL policies trained in calibrated EnergyPlus simulations can be deployed directly in real buildings.

Our environment suite includes a building generator that produces a virtually infinite number of environments derived from base archetypes across several geographic locations in North America. These environments expose low-level HVAC actuators such as fans and dampers, creating a wide variety of challenging control problems with heterogeneous observation and action spaces, and objectives reflecting different comfort-energy trade-offs. These models are automatically converted into Gymnasium-compatible RL environments through a dedicated processing pipeline.

Using this generator, we construct a dataset of 6000 environments spanning multiple building types, climates, and HVAC system configurations. We further propose well-defined and reproducible benchmark problems that isolate different sources of variation in RL: goal adaptation, dynamics variation, action-space shifts, and cross-domain generalization.

By providing a large and diverse collection of realistic control environments, *B2B* enables systematic and large-scale evaluation of generalization methods in RL. At the same time, it supports research on intelligent HVAC control and helps bridge the gap between fundamental RL research and real-world deployment on energy management systems. HVAC control represents an important societal challenge. Buildings account for roughly one third of global energy consumption, and HVAC systems alone are responsible for nearly half of this usage. Improving HVAC control can therefore reduce energy consumption, lower operational costs, and contribute directly to climate mitigation (IEA, 2022).

2 Related Work

2.1 Environments and Benchmarks for Generalization in RL

Existing benchmarks for studying generalization in RL are summarized in Table 1, and the different approaches to the generalization problem in RL are further discussed in Section A. Most of the benchmarks focus on either video games or robotics tasks such as locomotion and manipulation. As a result, they cover only a limited set of application domains. These problems have been extensively studied over the past decade. While they have led to significant algorithmic progress, it remains unclear whether these improvements generalize beyond the specific domains on which they are evaluated. As highlighted by Liao et al. (2021), repeatedly evaluating algorithms on the same benchmarks may lead to overfitting to the benchmark itself rather than genuine progress in general RL capabilities. This motivates the need for new environments covering different applications and offering large diversity of tasks.

Table 1: Main benchmarks used to study generalization in reinforcement learning.

Benchmark	Tasks	Domain
MineRL (Guss et al., 2019)	open / task generators	Video game (Minecraft)
Procgen (Cobbe et al., 2020)	16 task families	Procedural video games
RLBench (James et al., 2020)	100 tasks	Robotics manipulation
Meta-World (McLean et al., 2025)	50 tasks	Robotics manipulation
DM Control (Tunyasuvunakool et al., 2020)	∼20 tasks	Robotics locomotion / control
ManiSkill2 (Gu et al., 2023)	20 task families	Robotics manipulation
Robosuite (Zhu et al., 2020)	9 tasks	Robotics manipulation
MMBench (Hansen et al., 2026)	200 tasks	Games and robotics
Habitat (Szot et al., 2022)	open / configurable	Embodied navigation

2.2 HVAC control

Much of the existing literature on RL for HVAC follows a familiar pattern: select a single building, connect it to a simulator, and demonstrate that an agent can reduce energy consumption while maintaining thermal comfort. Surveys show that this type of study has become common practice (Sierla et al., 2022; Al Sayed et al., 2024). Optimal control of HVAC systems benefits from a rich literature, and Xu et al. (2025) propose to leverage expert knowledge to improve the sample efficiency of online methods. However, scaling online approaches beyond individual case studies remains difficult. Each new building requires its own digital twin, with sensors, actuators, and dynamics configured for simulation, and policies trained in one building rarely transfer to others.

Existing environments, summarized in Table 6, Section C reflect this limitation. Platforms such as Sinergym (Campoy-Nieves et al., 2025), Energym (Scharnhorst et al., 2021), BOPTEST (Blum et al., 2021), and CityLearn (Nweye et al., 2024) have accelerated research but collectively provide only a few dozen building models. Even within this limited set, policies trained in one environment often fail to generalize to others (Manjavacas et al., 2024). Transfer-learning approaches, such as fine-tuning across buildings (Kadamala et al., 2024; Xu et al., 2020; Coraci et al., 2024) or continual learning with hypernetworks (Bekal Udayakumar et al., 2025), provide incremental improvements but remain constrained by the lack of large-scale benchmarks. Field studies report similar challenges: a review of more than one hundred deployments finds inconsistent savings and limited reporting on integration costs (Khabbazi et al., 2025).

In contrast, *B2B* enables large-scale studies of RL for HVAC control. This capability enables a new training paradigm for HVAC control. Rather than designing a highly detailed digital twin for each individual building, policies can be trained across a distribution of buildings that includes the target building. Training across many environments reduces the need for highly accurate per-building models and allows policies to reuse experience across environments, reducing the need to train a new controller from scratch for every building.

3 The Building2Building Suite

3.1 Environment Generation

To access a diverse set of environments representative of real buildings, we rely on two complementary sources of building models that produce a large set of heterogeneous environments, summarized in Table 2[2].

- A dataset generated by a residential building generator (Larochelle Martin et al., 2026). This generator produces single-zone houses representing the statistical characteristics of the Québec residential building stock.

[2]Observation dimensions are given for constant-setpoint tasks; occupancy-based and random-setpoint tasks add one occupancy and one target-temperature signal per controlled zone.

Table 2: Base environments included in the benchmark suite.

Building type	HVAC type	Controlled zones	Action dim.	Observation dim.
House	Unitary system	1	2	9–11
Restaurant	Unitary systems	2	4	10
Warehouse	Unitary systems & baseboard	3	5	10
Retail Store	Unitary systems	5	9	12
Small Office	Unitary systems	5	10	13
Medium Office	Central systems	15	36	25

- A parametric building model generator based on the reference commercial buildings from the ASHRAE 90.1 across 16 different locations in North America. The generator follows the methodology used by the Building Technology Assessment Platform (Shirzadi, 2025), and samples the building size, window-to-wall ratio, insulation levels, and air infiltration. To ensure physical consistency, the generator enforces climate-dependent parameter bounds consistent with ASHRAE standards.

3.2 Control Interface

B2B exposes low-level HVAC actuators, and handles natively the following systems:

1. **Unitary HVAC systems.** These systems condition a single thermal zone using a dedicated unit such as a heat pump or packaged rooftop unit. For these systems, the agent controls the Supply Air Temperature (SAT) and the supply air flow rate.
2. **Central air loop systems for multi-zone conditioning.** These systems use centralized heating and cooling equipment that distributes conditioned air through an air loop serving multiple zones. The agent controls the central SAT, the outdoor-air mixer, the damper opening of each zone, and the zone-level reheat coils through thermostat setpoints.
3. **Baseboard and heating-only systems.** These systems provide heating only and are controlled through the zone thermostat.

At each timestep, the agent has access to the following observation: weather data (outdoor air temperature ($^\circ C$) and humidity ($\%$), calendar information (time of day, day of week, day of year), indoor zones air temperature ($^\circ C$), and HVAC energy consumption per simulation time step ($Wh/m^2/timestep$). Note that the energy consumption is normalized by the floor area of the building to be invariant to building size variations.

3.3 Control Tasks and Reward Functions

To study different trade-offs between thermal comfort and energy efficiency, we propose the following reward functions defining a family of control tasks parametrized by energy savings and comfort:

$$r_t = -\frac{1}{Z\,\tau_T}\sum_{z=1}^{Z}(T_z(t) - T_z^{target}(t))^2 - \frac{w_E}{\tau_E}\,E_{\text{HVAC}}(t), \tag{1}$$

where T_z is the zone temperature, T_z^{target} the zone target temperature, Z denotes the number of zones and E_{HVAC} the HVAC energy consumption per squared meter. Note that the target temperature T_z^{target} may vary over time depending on occupancy schedules.

Many widely used RL algorithms handle multiple objectives using a weighted sum of rewards (Hayes et al., 2022). The scale of each term in the sum defines the importance of the objective, and the weights have to be adjusted depending on the desired trade-off. Choosing appropriate weights is difficult and performance is highly sensitive to them (Lin et al., 2019). For objectives such as energy consumption for which the scale depends on the HVAC equipment and the building type, the

weights are environment dependent, and often tuned manually (Togashi, 2025). Defining the same trade-off with a single set of weights across thousands of environments is extremely challenging, if not impossible, and weights need to be adapted to each environment (van Hasselt et al., 2016; Hessel et al., 2018). To better handle the scale of each reward term, we introduce normalizing constants τ_T and τ_E that depend on the building type and the performances of a baseline policy. Using this normalized formulation, each term stays within the same order of magnitude across environments, and a single weight w_E may be used to define a comfort - energy savings trade-off across environments. More details about the reward formulation and normalizing constants are given in Section E.

3.4 Structured environment representation

A major obstacle in developing transfer and multi-task RL algorithms is the lack of standard ways to describe a heterogeneous fleet of different environments. Gymnasium's standard API, which presents an environment through its observation and action spaces, allows writing generic training algorithms and model architectures. However, no equivalent exists for representing families of environments whose action and observation spaces differ, but still form a cohesive family. Without a standard representation for the action and observation spaces, each algorithm for cross-domain transfer defines its own interface, specific to the domain of application. In an effort to facilitate cross-domain transfer research, *B2B* introduces a structured representation that provides a common abstraction for heterogeneous environments.

Rather than reinventing the environment interface, *B2B* extends it: an environment remains a standard Gymnasium environment, and its *morphology* is expressed as a pair of translation procedures between the observation spaces, the action spaces and a decomposed structured form. The decomposition separates each space in two parts: a *common* part shared by the set of environments, and a *morphological graph* carrying information that varies among the environments of the set. Departing from the usual two-space presentation of an environment, three types of data are tracked: observations, actions and *attributes*. *Attributes* are fixed for the lifetime of the environment and describe the environment domain rather than its current state. Attributes play the role of the context in a contextual MDP (Hallak et al., 2015): they identify which member of the family the agent is facing.

The vocabulary shared by all members of a family is collected in a *morphological universe*. Formally, a morphological universe U has the following structure:

1. A common attribute space C_c, a common observation space S_c and a common action space A_c.
2. A finite set of node types T.
3. For each node type $t \in T$, a local attribute space $C(t)$, a local observation space $S(t)$ and a local action space $A(t)$.

A universe describes a set of environments and a morphology describes one instance of environment. Given an environment (S, A, step), a morphology m is given by

1. common attributes $c \in C_c$;
2. a graph (V, E), together with a type $t(v) \in T$ and local attributes $c_v \in C(t(v))$ for each node $v \in V$.
3. an observation decomposition $\mathsf{split}\colon S \to S_c \times \prod_{v \in V} S(t(v))$;
4. an action recomposition $\mathsf{join}\colon A_c \times \prod_{v \in V} A(t(v)) \to A$.

This representation operates directly on the structure of the environment and naturally supports a large set of model architectures such as graph neural networks, heterogeneous graph Transformers (Hao et al., 2024) and type-heterogeneous encoder–decoder models (Ma et al., 2020). The separation between the morphological universe and the morphology also delineates clearly what a multi-morphology algorithm must be generic over, and when: on the morphological universe at algorithm instantiation time and on morphologies at specialization time.

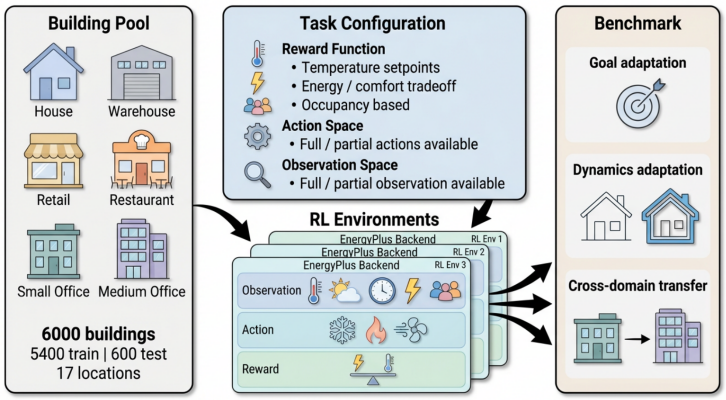


Figure 1: Overview of the benchmark design. A large building pool is created using our building generator. Different benchmark tasks evaluate transfer across goals, dynamics, and domains

B2B defines a single morphological universe and each building in the dataset provides a morphology in addition to the standard Gymnasium environment. In the *B2B* universe, quantities that exist once per building (outdoor weather, calendar information, and energy consumption) are modeled as dedicated observation-only node types rather than placed in the common spaces; the common attribute space is reserved for static building-level information such as the construction year. The remaining node types comprise thermal zones and the various types of HVAC equipment. The type "thermal zone" defines local attributes $C(t)$ that contain, for instance, the window-to-wall ratio.

4 Benchmark Problems

To define a reproducible experimental framework, we construct a diverse collection of environments by sampling buildings from the two generators described in Section 3. For each building type, we generate 1000 environments, resulting in a total dataset of 6000 environments, split into a training and a test set. The resulting environment suite spans a wide range of building dynamics, HVAC configurations, and control interfaces. Additional details on the environments generation are given in Section B.

Using these environments, we define four benchmark settings that isolate different sources of variation in control systems: goal adaptation, dynamics variation, action-space shifts, and cross-domain generalization. These benchmarks are easily accessible through the API interface of *B2B*.

- **Goal adaptation.** The training and test environments correspond to the same building and control interface, but the reward function changes. Six tasks are defined from the Cartesian product of the following predefined reward settings:
  - **Energy weight** $w_E \in \{0.0, 0.5\}$, corresponding to the absence or presence of an energy consumption penalty in the reward function. The choice of weights is discussed in Section E.
  - **Zone temperature setpoints** (1) constant setpoints ($21°C$); (2) occupancy-based setpoints ($21°C$ when occupied, $18°C$ or $26°C$ otherwise, depending on the season); (3) random setpoints[3] .
- **Dynamics adaptation - Table 3**. The reward function and action space remain fixed, while the building dynamics vary. Buildings differ in climate zone, size, and envelope but share the same archetype. The different settings have been chosen to vary the complexity of the dynamics : houses have a single thermal zone, small offices have multiple zones thermally coupled but with

[3]Random setpoints do not represent typical occupant behaviour. However, it increases the difficulty of the task as it de-correlates setpoints from the time of day and day of the week.

Table 3: Dynamics Adaptation Settings.

Setting	Building archetype	Variation	Action dim.
1	Single-zone house	Climate, building envelope parameters	2
2	Small office	Climate, building envelope parameters	10
3	Medium office	Climate, building envelope parameters	36

Table 4: Action Space Transfer Settings.

System type	Training control	Test control	Action dim.
Unitary system	Air flow rate	Air flow + SAT	$5 \rightarrow 10$
Central system	VAV boxes only	VAV boxes + central SAT	$33 \rightarrow 36$
Unitary system	Air flow + SAT	Air flow rate	$10 \rightarrow 5$
Central system	VAV boxes + central SAT	VAV boxes only	$36 \rightarrow 33$

independent HVAC systems, and medium offices have multiple zones thermally coupled with centralized HVAC systems in which some actions impact simultaneously multiple zones.

- **Action-space transfer - Table 4**. The building dynamics and reward function remain fixed, but the set of controllable actuators changes between training and testing.
- **Cross-domain generalization - Table 5**. Agents are trained and tested on different environments with different observation and action spaces. The predefined settings have been chosen to cover the different types of buildings and HVAC systems.

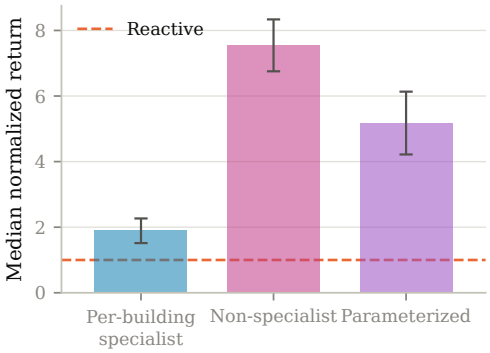
5 Baselines

To establish a baseline performance, we implement supervisory control logic inspired by the ASHRAE Guideline 36 high-performance sequences of operation. Two controllers are implemented: one for unitary systems and one for central air-loop systems. The control structures include zone-level PI loops that regulate airflow and Trim-and-Respond mechanisms that adjust the SAT. Additional implementation details are provided in Section D, and detailed control results are reported in Section F.1. Reactive control strategies are widely used in real buildings, and we therefore use them as the primary baseline. To facilitate evaluation, the benchmark includes a function *compute_normalized_score* that divides the cumulative return over a given period of simulation by the return obtained by the reactive controller, providing an easily interpretable normalized score.

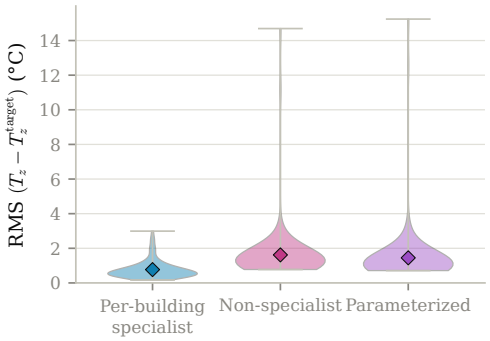
In addition, we train PPO agents on a subset of the test buildings for each building type. The subset contains 8 buildings, one per climate zone. For single-zone houses, which all belong to the same climate zone, 8 buildings are randomly sampled from the test set. Agents are trained on the four goal adaptation tasks defined in Section 4. Detailed results and normalized scores are reported in Section F.1.

Table 5: Cross-domain generalization settings

Setting	Train building type	Test building type	Key difference
1	Retail store	Small office	Similar HVAC types
2	Retail store	Warehouse	Slightly different HVAC types
3	Small office	Medium office	Different HVAC types
4	n types	m different types	Different domains



(a) Median normalized return on the test buildings. The dashed line represents the reactive control baseline performance.



(b) Violin plot of the RMS of temperature deviation from target temperature by model across test buildings. Diamonds represent the overall mean.

Figure 2: Performances of three types of models for dynamics adaptation

6 Generalization Experiments

6.1 Dynamics Adaptation

In order to assess the performance of current methods on dynamics adaptation, we consider the single-zone houses (Setting 1, Table 3). We train three different classes of models to track a constant setpoint with no penalty on the energy (i.e. $w_E = 0$) with PPO: (1) 100 per-building specialists that are trained solely on each building in the test set to establish an upper bound, (2) a non-specialist agent that resamples new buildings from the training set after every episode, and (3) a parameterized agent that resamples new buildings and has its observation space augmented with the building's floor area, year built, number of actuators and units. Implementation details are given in Section F.2.

Figure 2a shows the median normalized return for each type of model (with per-building results only on the building that model was trained on). As expected, the per-building specialist achieves the lowest (best) score, with a substantial gap from the parameterized and non-specialist models. The gap between the non-specialist and parameterized models shows the benefit of conditioning on building parameters for dynamics adaptation. As shown in Figure 2b, all models are able to control the temperature around the target. However, the non-specialist and parameterized models have a long tail distribution indicating a few cases among the 100 test buildings in which the transfer failed. On the other hand, per-building specialists performances are consistent throughout the test set.

6.2 Cross domain transfer

To demonstrate how *B2B* can be used to study cross-morphology control algorithms, we train a single policy simultaneously on four building types: retail store, fast food restaurant, small office and medium office. These types of buildings have heterogeneous action and observation spaces, making the environment incompatible with the standard multi-layer perceptron with fixed input and output sizes. We implement a variant of the Amorpheus (Kurin et al., 2021) architecture that we train on top of the structured environment representation described in Section 3.4.

The policy architecture consists of standard acausal transformer layers together with type-specific linear encoder and decoder blocks. At inference time, the observation is first split into node-level observations, which are encoded into a shared embedding space. The resulting embeddings are processed by the transformer, decoded into node-level actions, and finally concatenated to produce the global actuator command. Algorithm details are given in Section F.3.

The model is trained to track a dynamic setpoint while ignoring energy costs (i.e. $w_E = 0$). Training data are collected in parallel on four buildings (one of each type) during winter. Buildings are

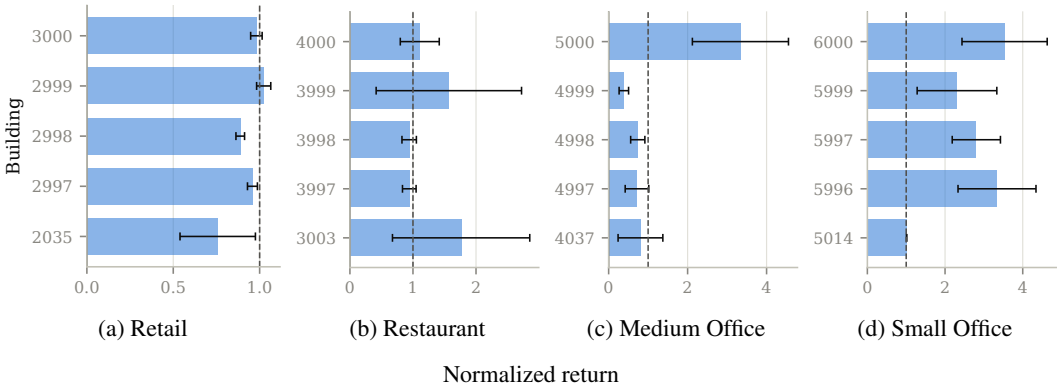


Figure 3: Generalized policy test performance on 20 unseen buildings sampled from the test set. Return is normalized by the reactive baseline on the same episode window (dashed line at 1.0); a value below 1 beats the baseline.

resampled every 10 PPO updates ensuring that the policy is exposed to the full diversity of buildings in the training set. The model is trained on 1M environment steps and tested on new buildings sampled from the test set. As shown in Figure 3 the policy is able to adapt to new unseen buildings, and even outperform the baseline reactive controller on some of them.

7 Conclusion

We introduce *B2B*, a large-scale suite of environments and benchmark problems designed to advance research on generalization in RL. *B2B* provides a diverse set of environments with varying dynamics and heterogeneous observation and action spaces, making it well suited for studying multi-task learning and cross-domain transfer. The benchmark tasks cover several forms of adaptation, including goal adaptation and action-space shifts, and provide a reproducible framework for comparing algorithms.

Grounded in real-world applications, the control tasks focus on HVAC systems in buildings. Improving control in this domain has the potential to reduce energy consumption and contribute significantly to climate change mitigation.

Acknowledgments

This research was enabled in part by compute resources provided by Mila (mila.quebec).

References

Ayah Al Sayed, Yasser Abouelatta, and Mostafa Elshafei. Applications of reinforcement learning in hvac control systems: A systematic review. *Energies*, 15(10):3526, 2024.

Andre Barreto, Will Dabney, Remi Munos, Jonathan J Hunt, Tom Schaul, Hado P van Hasselt, and David Silver. Successor features for transfer in reinforcement learning. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/350db081a661525235354dd3e19b8c05-Paper.pdf.

Nithin Bekal Udayakumar, Ali Esmaili, Varun Gupta, et al. Continual learning for hvac control: A hypernetwork-based reinforcement learning approach. *arXiv preprint arXiv:2503.19212*, 2025.

David Blum, Javier Arroyo, Sen Huang, Ján Drgoňa, Filip Jorissen, Harald Taxt Walnum, Yan Chen, Kyle Benne, Draguna Vrabie, Michael Wetter, and Lieve Helsen. Building optimization

testing framework (boptest) for simulation-based benchmarking of control strategies in buildings. *Journal of Building Performance Simulation*, 14(5):586–610, 2021. DOI: 10.1080/19401493.2021.1986574. URL https://doi.org/10.1080/19401493.2021.1986574.

Alejandro Campoy-Nieves, Antonio Manjavacas, Javier Jiménez-Raboso, Miguel Molina-Solana, and Juan Gómez-Romero. Sinergym – a virtual testbed for building energy optimization with reinforcement learning. *Energy and Buildings*, 327, 2025. ISSN 0378-7788. DOI: 10.1016/j.enbuild.2024.115075. URL https://www.sciencedirect.com/science/article/pii/S0378778824011915.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Michael Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling, 2021. URL https://arxiv.org/abs/2106.01345.

Karl Cobbe, Christopher Hesse, Jacob Hilton, and John Schulman. Leveraging procedural generation to benchmark reinforcement learning, 2020. URL https://arxiv.org/abs/1912.01588.

Daniele Coraci, Maria Di Francesco, Alessio Angioni, et al. Multi-source transfer learning for multi-zone building hvac control with deep reinforcement learning. *Building Simulation*, 17(2): 233–247, 2024.

Peter Dayan. Improving generalization for temporal difference learning: The successor representation. *Neural Computation*, 5(4):613–624, 1993. DOI: 10.1162/neco.1993.5.4.613.

Yan Duan, John Schulman, Xi Chen, Peter L. Bartlett, Ilya Sutskever, and Pieter Abbeel. Rl^2: Fast reinforcement learning via slow reinforcement learning, 2016. URL https://arxiv.org/abs/1611.02779.

Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Volodymir Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, Shane Legg, and Koray Kavukcuoglu. Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures, 2018. URL https://arxiv.org/abs/1802.01561.

Arnaud Fickinger, Samuel Cohen, Stuart Russell, and Brandon Amos. Cross-domain imitation learning via optimal transport. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=xP3cPq2hQC.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1126–1135. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/finn17a.html.

Judah Goldfeder, Victoria Dean, Zixin Jiang, Xuezheng Wang, Bing dong, Hod Lipson, and John Sipple. The smart buildings control suite: A diverse open source benchmark to evaluate and scale hvac control policies for sustainability, 2025. URL https://arxiv.org/abs/2410.03756.

Jiayuan Gu, Fanbo Xiang, Xuanlin Li, Zhan Ling, Xiqiang Liu, Tongzhou Mu, Yihe Tang, Stone Tao, Xinyue Wei, Yunchao Yao, Xiaodi Yuan, Pengwei Xie, Zhiao Huang, Rui Chen, and Hao Su. Maniskill2: A unified benchmark for generalizable manipulation skills, 2023. URL https://arxiv.org/abs/2302.04659.

Agrim Gupta, Silvio Savarese, Surya Ganguli, and Li Fei-Fei. Embodied intelligence via learning and evolution. *Nature Communications*, 12(1), October 2021. ISSN 2041-1723. DOI: 10.1038/s41467-021-25874-z. URL http://dx.doi.org/10.1038/s41467-021-25874-z.

Agrim Gupta, Linxi Fan, Surya Ganguli, and Li Fei-Fei. Metamorph: Learning universal controllers with transformers, 2022. URL https://arxiv.org/abs/2203.11931.

William H. Guss, Brandon Houghton, Nicholay Topin, Phillip Wang, Cayden Codel, Manuela Veloso, and Ruslan Salakhutdinov. Minerl: A large-scale dataset of minecraft demonstrations, 2019. URL https://arxiv.org/abs/1907.13440.

Assaf Hallak, Dotan Di Castro, and Shie Mannor. Contextual markov decision processes, 2015. URL https://arxiv.org/abs/1502.02259.

Nicklas Hansen, Hao Su, and Xiaolong Wang. Learning massively multitask world models for continuous control. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=MPabX9LEds.

YiFan Hao, Yang Yang, Junru Song, Wei Peng, Weien Zhou, Tingsong Jiang, and Wen Yao. Heteromorpheus: Universal control based on morphological heterogeneity modeling, 2024. URL https://arxiv.org/abs/2408.01230.

Conor F. Hayes, Roxana Rădulescu, Eugenio Bargiacchi, Johan Källström, Matthew Macfarlane, Mathieu Reymond, Timothy Verstraeten, Luisa M. Zintgraf, Richard Dazeley, Fredrik Heintz, Enda Howley, Athirai A. Irissappane, Patrick Mannion, Ann Nowé, Gabriel Ramos, Marcello Restelli, Peter Vamplew, and Diederik M. Roijers. A practical guide to multi-objective reinforcement learning and planning. *Autonomous Agents and Multi-Agent Systems*, 36(1):26, 2022. DOI: 10.1007/s10458-022-09552-y.

Ahmed Hendawy, Jan Peters, and Carlo D'Eramo. Multi-task reinforcement learning with mixture of orthogonal experts. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=aZH1dM3GOX.

Matteo Hessel, Hubert Soyer, Lasse Espeholt, Wojciech Czarnecki, Simon Schmitt, and Hado van Hasselt. Multi-task deep reinforcement learning with popart, 2018. URL https://arxiv.org/abs/1809.04474.

IEA. Iea world energy outlook report 2022. https://www.iea.org/reports/world-energy-outlook-2022, 2022.

Stephen James, Zicong Ma, David Rovick Arrojo, and Andrew J. Davison. Rlbench: The robot learning benchmark & learning environment. *IEEE Robotics and Automation Letters*, 5(2):3019–3026, 2020. DOI: 10.1109/LRA.2020.2974707.

Kevlyn Kadamala, Des Chambers, and Enda Barrett. Enhancing hvac control systems through transfer learning with deep reinforcement learning agents. *Smart Energy*, 13:100131, 2024. ISSN 2666-9552. DOI: https://doi.org/10.1016/j.segy.2024.100131. URL https://www.sciencedirect.com/science/article/pii/S2666955224000017.

Maryam Khabbazi, Rens van der Meer, Glenn Reynders, and Lieve Helsen. Methodological shortcomings in field demonstrations of advanced building control: A systematic review. *arXiv preprint arXiv:2503.05022*, 2025.

Vitaly Kurin, Maximilian Igl, Tim Rocktäschel, Wendelin Boehmer, and Shimon Whiteson. My body is a cage: the role of morphology in graph-based incompatible control, 2021. URL https://arxiv.org/abs/2010.01856.

Gilbert Larochelle Martin, Benoit Delcroix, Brice Le Lostec, Seyedsaeid Hosseini, Aziz Mbaye, Mike Coillot, and Simon Sansregret. Residential building stock model of the province of Quebec, Canada: Methodology & preliminary results. In *Proceedings of eSim 2026: 14th Conference of IBPSA-Canada*, Longueuil, QC, Canada, June 2026. IBPSA-Canada.

Michael Laskin, Luyu Wang, Junhyuk Oh, Emilio Parisotto, Stephen Spencer, Richie Steigerwald, DJ Strouse, Steven Hansen, Angelos Filos, Ethan Brooks, Maxime Gazeau, Himanshu Sahni, Satinder Singh, and Volodymyr Mnih. In-context reinforcement learning with algorithm distillation, 2022. URL https://arxiv.org/abs/2210.14215.

Jonathan Lee, Annie Xie, Aldo Pacchiano, Yash Chandak, Chelsea Finn, Ofir Nachum, and Emma Brunskill. Supervised pretraining can learn in-context reinforcement learning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=dCYBAGQXLo.

Thomas Liao, Rohan Taori, Deborah Raji, and Ludwig Schmidt. Are we learning yet? a meta review of evaluation failures across machine learning. In J. Vanschoren and S. Yeung (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/file/757b505cfd34c64c85ca5b5690ee5293-Paper-round2.pdf.

Xi Lin, Hui-Ling Zhen, Zhenhua Li, Qingfu Zhang, and Sam Kwong. Pareto multi-task learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

Chao Ma, Sebastian Tschiatschek, Richard Turner, José Miguel Hernández-Lobato, and Cheng Zhang. Vaem: a deep generative model for heterogeneous mixed type data. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 11237–11247. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/8171ac2c5544a5cb54ac0f38bf477af4-Paper.pdf.

Álvaro Manjavacas, Alberto Campoy-Nieves, Javier García-Gutiérrez, and Francisco García. Experimental evaluation of reinforcement learning algorithms for energy-efficient hvac control in sinergym. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2024. arXiv:2401.05737.

Reginald McLean, Evangelos Chatzaroulas, Luc McCutcheon, Frank Röder, Tianhe Yu, Zhanpeng He, K.R. Zentner, Ryan Julian, J K Terry, Isaac Woungang, Nariman Farsad, and Pablo Samuel Castro. Meta-world+: An improved, standardized, RL benchmark. In *Championing Open-source DEvelopment in ML Workshop @ ICML25*, 2025. URL https://openreview.net/forum?id=eYZ9ebLIXo.

Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. A simple neural attentive meta-learner, 2018. URL https://arxiv.org/abs/1707.03141.

Takao Moriyama, Giovanni De Magistris, Michiaki Tatsubori, Tu-Hoa Pham, Asim Munawar, and Ryuki Tachibana. Reinforcement learning testbed for power-consumption optimization. In *Methods and Applications for Modeling and Simulation of Complex Systems*, pp. 45–59, Singapore, 2018. Springer Singapore. ISBN 978-981-13-2853-4.

Michal Nauman, Marek Cygan, Carmelo Sferrazza, Aviral Kumar, and Pieter Abbeel. Bigger, regularized, categorical: High-capacity value functions are efficient multi-task learners. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=zhOUfuOIzA.

NREL. Energyplus™, 09 2017. URL https://www.osti.gov/biblio/1395882.

Kingsley Nweye, Kathryn Kaspar, Giacomo Buscemi, Tiago Fonseca, Giuseppe Pinto, Dipanjan Ghose, Satvik Duddukuru, Pavani Pratapa, Han Li, Javad Mohammadi, Luis Lino Ferreira, Tianzhen Hong, Mohamed Ouf, Alfonso Capozzoli, and Zoltan Nagy. Citylearn v2: energy-flexible, resilient, occupant-centric, and carbon-aware management of grid-interactive communities. *Journal of Building Performance Simulation*, 0(0):1–22, 2024. DOI: 10.1080/19401493.2024.2418813. URL https://doi.org/10.1080/19401493.2024.2418813.

Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8, 2021. URL http://jmlr.org/papers/v22/20-1364.html.

Kate Rakelly, Aurick Zhou, Deirdre Quillen, Chelsea Finn, and Sergey Levine. Efficient off-policy meta-reinforcement learning via probabilistic context variables, 2019. URL https://arxiv.org/abs/1903.08254.

Paul Scharnhorst, Baptiste Schubnel, Carlos Fernández Bandera, Jaume Salom, Paolo Taddeo, Max Boegli, Tomasz Gorecki, Yves Stauffer, Antonis Peppas, and Chrysa Politi. Energym: A building model library for controller benchmarking. *Applied Sciences*, 11(8), 2021. ISSN 2076-3417. DOI: 10.3390/app11083518. URL https://www.mdpi.com/2076-3417/11/8/3518.

Navid Shirzadi. Surrogate modeling for building design: Energy and cost prediction compared to simulation-based methods. *Buildings*, 15(13):2361, 2025. DOI: 10.3390/buildings15132361. URL https://www.mdpi.com/2075-5309/15/13/2361.

Seppo Sierla, Asko Huuskonen, Jarmo Kuusisto, and Valeriy Vyatkin. Reinforcement learning in building energy management: A review. *Applied Energy*, 322:119–169, 2022.

Andrew Szot, Alex Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Chaplot, Oleksandr Maksymets, Aaron Gokaslan, Vladimir Vondrus, Sameer Dharur, Franziska Meier, Wojciech Galuba, Angel Chang, Zsolt Kira, Vladlen Koltun, Jitendra Malik, Manolis Savva, and Dhruv Batra. Habitat 2.0: Training home assistants to rearrange their habitat, 2022. URL https://arxiv.org/abs/2106.14405.

Yee Whye Teh, Victor Bapst, Wojciech Marian Czarnecki, John Quan, James Kirkpatrick, Raia Hadsell, Nicolas Heess, and Razvan Pascanu. Distral: Robust multitask reinforcement learning, 2017. URL https://arxiv.org/abs/1707.04175.

Eisuke Togashi. Reward function design in reinforcement learning for HVAC control: A review of thermal comfort and energy efficiency trade-offs. *Energy and Buildings*, 348:116439, 2025. DOI: 10.1016/j.enbuild.2025.116439.

Mark Towers, Ariel Kwiatkowski, Jordan Terry, John U. Balis, Gianluca De Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Markus Krimmel, Arjun KG, Rodrigo Perez-Vicente, Andrea Pierré, Sander Schulhoff, Jun Jet Tai, Hannah Tan, and Omar G. Younis. Gymnasium: A standard interface for reinforcement learning environments, 2024. URL https://arxiv.org/abs/2407.17032.

Brandon Trabucco, Mariano Phielipp, and Glen Berseth. AnyMorph: Learning transferable polices by inferring agent morphology. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 21677–21691. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/trabucco22b.html.

Saran Tunyasuvunakool, Alistair Muldal, Yotam Doron, Siqi Liu, Steven Bohez, Josh Merel, Tom Erez, Timothy Lillicrap, Nicolas Heess, and Yuval Tassa. dm_control: Software and tasks for continuous control. *Software Impacts*, 6:100022, 2020. ISSN 2665-9638. DOI: 10.1016/j.simpa.2020.100022. URL https://www.sciencedirect.com/science/article/pii/S2665963820300099.

Hado van Hasselt, Arthur Guez, Matteo Hessel, Volodymyr Mnih, and David Silver. Learning values across many orders of magnitude. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 29, 2016.

Qingyang Xu, Yanzi Yin, Xiangyu Li, et al. One-for-many: Transfer learning for building hvac control under varying configurations. *arXiv preprint arXiv:2008.03625*, 2020.

Shichao Xu, Yangyang Fu, Yixuan Wang, Zhuoran Yang, Chao Huang, Zheng O'Neill, Zhaoran Wang, and Qi Zhu. Efficient and assured reinforcement learning-based building hvac control with heterogeneous expert-guided training. *Scientific Reports*, 15(1):7677, 2025. ISSN 2045-2322. DOI: 10.1038/s41598-025-91326-z. URL https://doi.org/10.1038/s41598-025-91326-z.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 5824–5836. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/3fe78a8acf5fda99de95303940a2420c-Paper.pdf.

Zhiang Zhang, Adrian Chong, Yuqi Pan, Chenlu Zhang, and Khee Poh Lam. Whole building energy model for hvac optimal control: A practical framework based on deep reinforcement learning. *Energy and Buildings*, 199:472–490, 2019. ISSN 0378-7788. DOI: https://doi.org/10.1016/j.enbuild.2019.07.029. URL https://www.sciencedirect.com/science/article/pii/S0378778818330858.

Yuke Zhu, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Abhishek Joshi, Soroush Nasiriany, Yifeng Zhu, and Kevin Lin. robosuite: A modular simulation framework and benchmark for robot learning. In *arXiv preprint arXiv:2009.12293*, 2020.

Luisa Zintgraf, Kyriacos Shiarlis, Maximilian Igl, Sebastian Schulze, Yarin Gal, Katja Hofmann, and Shimon Whiteson. Varibad: A very good method for bayes-adaptive deep rl via meta-learning, 2020. URL https://arxiv.org/abs/1910.08348.

Supplementary Materials

The following content was not necessarily subject to peer review.

A Prior work on generalization in RL

Generalization in RL has been widely studied. A common approach is Meta RL, where an agent is trained on a distribution of tasks so it can quickly adapt to new ones. Early methods include Model-Agnostic Meta-Learning (MAML) (Finn et al., 2017), which adapts policies with a few gradient steps, and RL^2 (Duan et al., 2016), which embeds the learning algorithm inside a recurrent network to share information across tasks via the RNN's hidden state. Later work introduced adaptation mechanisms such as attention-based meta-learners (Mishra et al., 2018), and probabilistic context inference methods such as PEARL (Rakelly et al., 2019), and Bayes-Adaptive RL (VariBAD) (Zintgraf et al., 2020).

A related line of work studies generalization through a multi-task formulation, where a single model is trained on multiple tasks simultaneously. Such an approach is challenging because gradients from different tasks interfere during learning. Methods such as PCGrad (Yu et al., 2020) mitigate gradient conflicts between tasks during training, while other approaches rely on multiple learner, such as policy distillation (Distral) (Teh et al., 2017) and mixture of orthogonal experts (Hendawy et al., 2024). Large-scale systems such as IMPALA (Espeholt et al., 2018) and BRC (Nauman et al., 2025) demonstrated positive transfer across many tasks, while normalization techniques such as PopArt help address reward-scale imbalance across tasks(Hessel et al., 2018).

Recent works also studies adaptation using sequence modeling and in-context reinforcement learning. In these approaches, models are trained on large datasets of trajectories and adapt to new tasks through conditioning on past experience in context. Seminal works in this direction include Decision Transformer, (Chen et al., 2021), Decision Pretrained Transformer (Lee et al., 2023) and Algorithm Distillation (Laskin et al., 2022).

Most multi-task RL methods assume shared observation and action spaces across tasks. However, real-world problems often involve changes in morphology or environments. Methods such as DERL (Gupta et al., 2021), MetaMorph (Gupta et al., 2022), and AnyMorph (Trabucco et al., 2022) study control across agents with different morphologies. Several works study theoretically grounded transfer across domains. For instance successor representations and successor features enable reuse of value structure across related tasks (Barreto et al., 2017; Dayan, 1993). Other approaches provide principled alignment across domains, for example through optimal-transport-based imitation learning (Fickinger et al., 2022).

B Building dataset generation

We generate a climate-consistent building dataset using climate-dependent parameter ranges derived from ASHRAE 90.1-2022 Tables 5.5-1 through 5.5-8. Instead of sampling all seven envelope and geometry parameters independently over uniform global bounds, each building's location is first mapped to its ASHRAE climate zone, and LHS unit samples are then remapped to zone-specific ranges defined relative to the prescriptive maxima for fenestration and framed walls. Window U-factors are sampled within $[0.8,\ 1.3]\times$ the ASHRAE maximum for the given zone; SHGC is sampled between 0.15 and $(\text{SHGC}_{\max} + 0.10)$, clamped to $[0.10,\ 0.80]$; and envelope conductivity and infiltration multipliers are upper-bounded as a function of climate stringency, ensuring tighter constructions in colder zones (ASHRAE 90.1-2022, §5.4.3.1; Tables 5.5-1–5.5-8). Geometry scaling is narrowed to $[0.7,\ 1.5]$ to avoid extreme HVAC autosizing artifacts, while orientation remains uniformly sampled in $[0^\circ,\ 360^\circ]$. LHS sampling is preserved within each climate zone by first generating unit hypercube samples and then mapping them through zone-specific bounds. Finally, balanced train/test splits are produced with a dedicated script that performs stratified sampling by location (90/10), ensuring proportional climate representation across splits.

Table 6: Comparison of building control environments

Environment	Buildings
RL Testbed (Moriyama et al., 2018)	1
BOPTEST(-Gym) (Blum et al., 2021)	∼12
Energym (Scharnhorst et al., 2021)	14
Sinergym (Campoy-Nieves et al., 2025)	∼30
Smart bldg. Control (Goldfeder et al., 2025)	11
CityLearn (Nweye et al., 2024)	Variable
Building2Building (Ours)	**Thousands**

C Environments details

Building2Building converts building simulation models into standardized RL environments. Starting from building descriptions encoded as EnergyPlus input files, *B2B* automatically constructs a controllable simulation environment compatible with the Gymnasium API (Towers et al., 2024). The pipeline parses the building model, identifies thermal zones and HVAC systems, exposes relevant control actuators, and generates standardized observation variables. This process abstracts away simulator-specific details and produces environments with consistent interfaces for observations, actions, and rewards. As a result, RL agents can be evaluated across a diverse set of buildings without requiring any simulator-specific implementations.

The resulting Gymnasium environment interacts with the EnergyPlus simulation using callbacks at every step to fetch observations and pass actions. An EnergyPlus timestep takes about 0.1ms, and multiple EnergyPlus simulations can be launched in parallel. Timestep durations for the different building types are given in Table 7.

Other HVAC control environments based on high-fidelity open-source models, such as Sinergym (Campoy-Nieves et al., 2025), provide only a limited number of buildings. This limitation arises from two main challenges. First, EnergyPlus simulations require detailed input building descriptions, which are rarely available at large scale. Second, interacting with a complex simulator such as EnergyPlus in a consistent way is difficult, especially when exposing low-level HVAC actuators for control. *B2B* addresses these challenges in two ways. First, it provides a generator of EnergyPlus building files that enables the creation of numerous building environments in a principled way, allowing sensors and actuators to be extracted automatically. Second, it includes a processing pipeline that automatically converts EnergyPlus models into Python RL environments following the Gymnasium interface, allowing direct use with existing RL algorithms.

Table 7: EnergyPlus simulation step time per building type, computed over 10000 steps

Building Type	Mean (ms/step)	Std (ms/step)
Office Small	0.170	0.018
Office Medium	0.410	0.096
Retail Standalone	0.153	0.021
Restaurant Fast Food	0.059	0.006
Warehouse	0.209	0.044
Single-Family House	0.021	0.002

D Baseline controllers

D.1 Reactive Control Logic for Unitary Systems

For zones equipped with a unitary system, the action space is the supply fan air mass flow rate and the supply air temperature setpoint. The control logic follows supervisory principles of ASHRAE Guideline-36, adapted to a 5-minute timestep. The zone temperature is regulated using a PI controller that modulates the supply air flow rate. Let T_z denote the zone air temperature and T_{sp} the active temperature setpoint (cooling or heating). The control error is defined as

$$e(t) = T_z(t) - T_{sp}(t). \tag{2}$$

The airflow command is

$$\dot{m}_{sa}(t) = \dot{m}_{\min} + \left[K_p e(t) + K_i \int_0^t e(\tau)\, d\tau \right] (\dot{m}_{\max} - \dot{m}_{\min}), \tag{3}$$

subject to saturation:

$$\dot{m}_{sa} \in [\dot{m}_{\min}, \dot{m}_{\max}]. \tag{4}$$

The gains K_p and K_i are tuned for stability under a 5-minute timestep. When the zone temperature lies within the deadband, airflow is reduced to $\dot{m}_{\min}$.

The supply air temperature setpoint is adjusted using a Trim-and-Respond logic based on the zone demand. For instance, in a cooling regime:

$$T_{sa,sp}^{k+1} = \begin{cases} T_{sa,sp}^{k} - \Delta_{\text{resp}}, & \text{if } e_c^k > \delta \\ T_{sa,sp}^{k} + \Delta_{\text{trim}}, & \text{otherwise} \end{cases} \tag{5}$$

subject to bounds:

$$T_{sa,sp} \in [T_{sa,\min},\, T_{sa,\max}]. \tag{6}$$

Here Δ_{resp} and Δ_{trim} represent the respond and trim increments, respectively.

D.2 Reactive Control Logic for Air Loops

For zones served by a central air loop, the action space consists of the central supply air temperature setpoint $T_{sa,sp}$ and the per-zone damper opening fraction $\alpha_i \in [0, 1]$. Each zone i is equipped with a thermostat that modulates the local reheat coil. The zone temperature is regulated through the damper position using a PI controller. Let $T_{z,i}$ denote the zone air temperature and $T_{sp,i}$ the active temperature setpoint. The control error is

$$e_i(t) = T_{z,i}(t) - T_{sp,i}(t). \tag{7}$$

The commanded damper opening fraction is

$$\alpha_i(t) = \alpha_{\min} + \left[K_p e_i(t) + K_i \int_0^t e_i(\tau)\, d\tau \right] (\alpha_{\max} - \alpha_{\min}), \tag{8}$$

subject to

$$\alpha_i \in [\alpha_{\min}, \alpha_{\max}]. \tag{9}$$

When the zone temperature lies within the deadband, the damper position is maintained at $\alpha_{\min}$.

Each zone thermostat activates the reheat coil when the zone temperature falls below the heating setpoint. The reheat power $Q_{rh,i}$ is modulated proportionally to the heating error

$$Q_{rh,i}(t) = K_{rh}\,[T_{sp,i}(t) - T_{z,i}(t)]_+, \tag{10}$$

where $[x]_+ = \max(0, x)$.

The central supply air temperature setpoint is adjusted using a Trim-and-Respond logic based on aggregate cooling demand:

$$T_{sa,sp}^{k+1} = \begin{cases} T_{sa,sp}^{k} - \Delta_{\text{resp}}, & \text{if } D_c^k > \delta \\ T_{sa,sp}^{k} + \Delta_{\text{trim}}, & \text{otherwise} \end{cases} \tag{11}$$

subject to

$$T_{sa,sp} \in [T_{sa,\min}, T_{sa,\max}]. \tag{12}$$

D.3 PPO implementation

We use Stable-Baselines3 (Raffin et al., 2021) for the PPO implementation. Rollouts are collected on 14 environments in parallel. The training is done over 5M environment steps. The hyperparameters used are summarized in Table 8.

Table 8: Hyperparameters used for PPO training.

Parameter	**Value**	**Parameter**	**Value**
Policy type	MLP	Learning rate	5×10^{-5}
Batch size	336	*n_steps*	672
Target KL	0.02	*n_epochs*	5
γ	0.98	λ (GAE)	0.95
Clip range	0.2	Entropy coef.	0.01
Value function coef.	0.5	Max grad norm	0.5
Use SDE	False	SDE sample freq.	96
Policy network architecture			
Actor layers	[256, 256]	Critic layers	[256, 256]
Activation	Tanh	Orthogonal init	True
Initial log std	-1.0		

E Reward normalization

Recall the reward definition given by (Equation (1)):

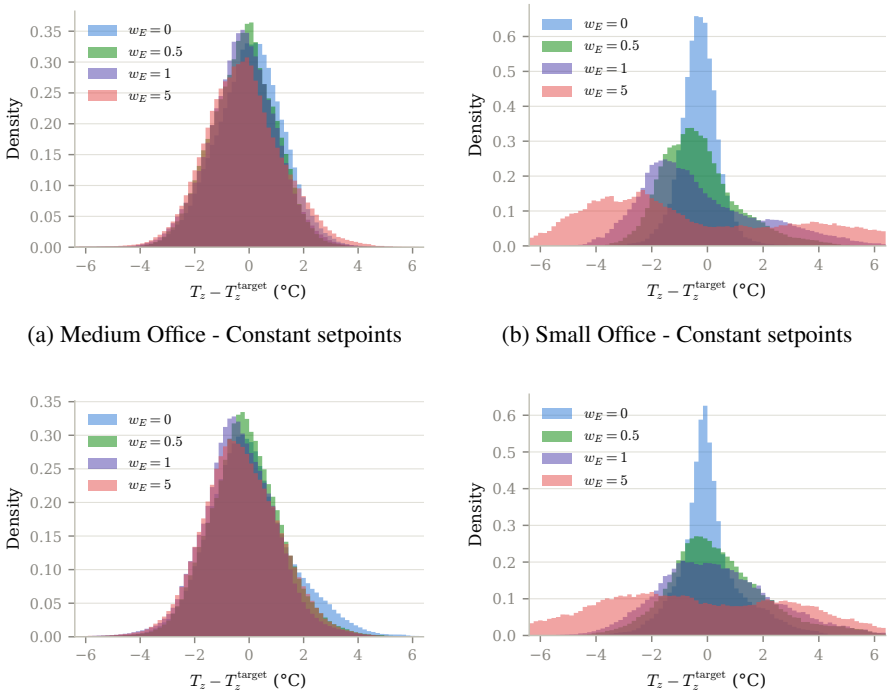
$$r_t = -\frac{1}{Z\,\tau_T}\sum_{z=1}^{Z}(T_z(t) - T_z^{target}(t))^2 - \frac{w_E}{\tau_E}\,E_{\text{HVAC}}(t).$$

A difference of $1^\circ C$ in zone temperature has the same meaning across buildings and is easily interpretable, we thus fix $\tau_T = 1$. The energy consumption term however depends on the building type,

the building size, the HVAC type and the climate zone. To normalize the scale of the reward, we fix τ_E as the mean per-step power consumption of the baseline reactive policy of a given building type and climate zone. τ_E is building and climate dependent, and allows controlling the scale of the energy penalty term in the sum. $E_{\text{HVAC}}(t)/\tau_E$ is dimensionless and w_E has a physical interpretation : how much temperature deviation (in $°C$) is worth a unit of energy consumption (with respect to the baseline power consumption).

We investigate the behaviour of learned PPO policies with respect to the choice of $w_E \in \{0; 0.5; 1.0; 5.0\}$ on two building types, as shown in Figure 4 and Figure 5. As expected, the higher w_E the lower the total energy consumption. However, despite the normalization, we observe different impacts of w_E on the temperature control : in the Medium office, the mean temperature deviation distribution does not shift much, but in the Small office for high values of w_E the learned policy is degenerate and focus only on limiting power consumption by turning off the HVAC system. This highlights that the same choice of weights does not necessarily result in the same objectives trade-off depending on the environment.

The pre-defined tasks feature two choices of rewards : temperature control only ($w_E = 0$) and with an energy consumption penalty ($w_E = 0.5$). We kept $w_E = 0.5$ as it yields no degenerate policies on every building and climate zones of the small test set in both winter and summer periods.



(a) Medium Office - Constant setpoints

(b) Small Office - Constant setpoints

(c) Medium Office - Occupancy based setpoints

(d) Small Office - Occupancy based setpoints

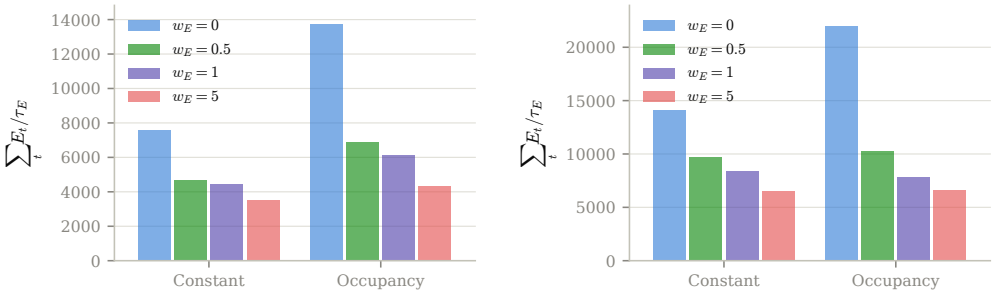
Figure 4: Temperature distribution for different energy penalty weights

F Additional experiments

F.1 Baselines results

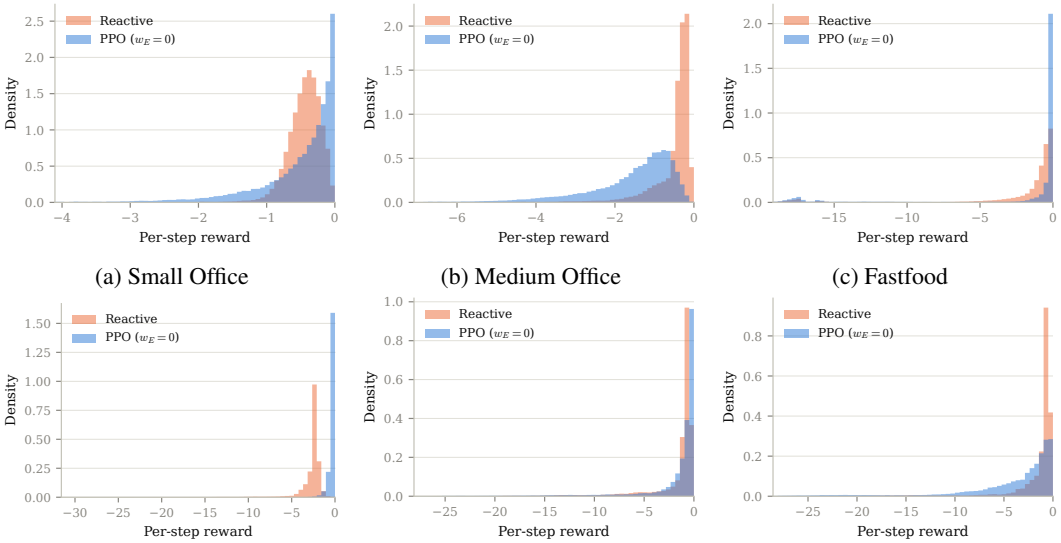
In this section we present the results of the reactive controller and PPO agents on the test buildings.

For the reactive controllers, hyperparameters of each controller were tuned on the test buildings using Bayesian optimization to optimize for 25^{th} percentile performance, with occupancy based



(a) Medium Office (b) Small Office

Figure 5: Energy consumption for different energy penalty weights



(a) Small Office (b) Medium Office (c) Fastfood

(d) House (e) Warehouse (f) Retail

Figure 6: Per step reward distribution aggregated over the 8 buildings of the small test set for constant setpoints.

setpoints and no energy penalty as the optimization objective. One set of hyperparameters is used per building type and climate zone. Figure 6 and Figure 7 present the performance of the reactive controllers. Reactive controllers are designed for temperature control only, we thus fix $w_E = 0$ and the reward represents temperature deviation (in $°C$). The distributions' modes are around 0, indicating a good reward control. We explain the tail of the distribution by the fixed set of hyperparameters. Indeed, some buildings such as small buildings with low thermal inertia are more sensitive to specific choices of hyperparameter.

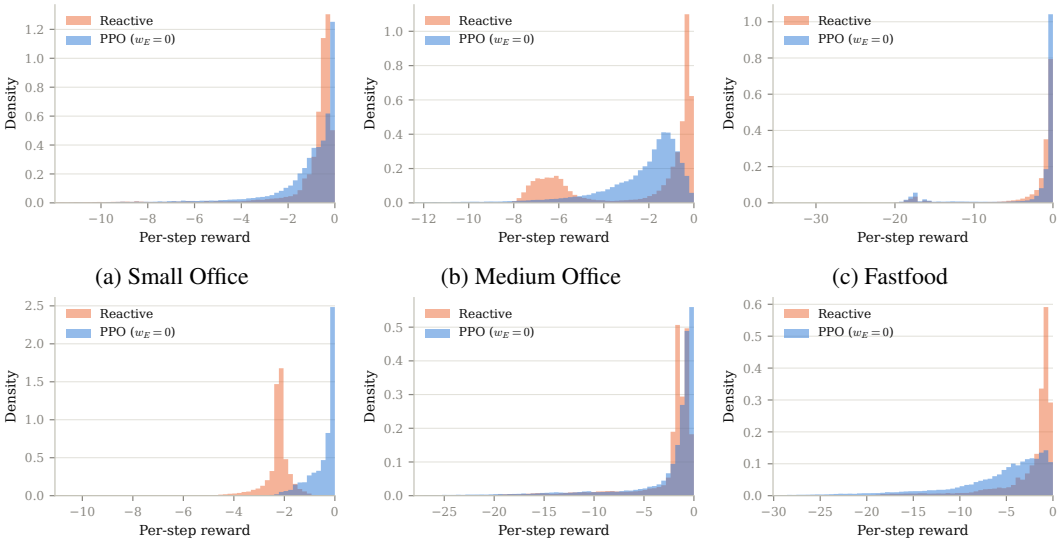
PPO agents are trained specifically on each building of the small test set, across different tasks and seasons, using the hyperparameters of Section D.3. Table 9 shows the results of the Figure 6 and Figure 7 compares the per-step reward distribution of the PPO agents to the reactive control baseline.

F.2 Dynamics adaptation

All buildings in this setting have a two-dimensional action space. Their observation spaces are padded to 20 dimensions, as the number of unconditioned zones (and hence of observed zone temperatures) varies across buildings. Episodes span 90 days of the winter period, i.e., 25,920 timesteps

Table 9: Performances of PPO agents on the small test set for different tasks and seasons. The scores are normalized with respect to the reactive baseline performance. The *Beats* columns indicates the number of test buildings in which PPO outperforms the baseline.

			Winter				Summer			
Building type	Setpoint	w_E	Median	Min	Max	Beats	Median	Min	Max	Beats
Office Small	Constant	0	0.95	0.54	2.78	4/8	0.47	0.19	0.53	8/8
		0.5	2.09	1.24	4.94	0/8	1.05	0.57	2.07	2/8
	Occupancy	0	1.19	0.62	1.52	2/8	1.35	0.37	1.89	1/8
		0.5	1.29	0.76	10.86	1/8	1.89	1.51	3.36	0/8
Office Medium	Constant	0	2.80	1.49	9.07	0/8	1.65	0.60	4.09	3/8
		0.5	1.58	0.93	2.35	1/8	0.95	0.59	2.03	4/8
	Occupancy	0	1.04	0.57	1.32	4/8	0.31	0.25	0.39	8/8
		0.5	0.72	0.52	0.89	8/8	0.30	0.25	0.39	8/8
Retail Standalone	Constant	0	2.42	1.42	3.79	0/8	2.62	1.11	9.08	0/8
		0.5	1.97	1.55	7.87	0/8	2.98	1.13	10.10	0/8
	Occupancy	0	2.28	1.54	5.01	0/8	2.29	0.62	3.19	1/8
		0.5	2.43	1.26	3.94	0/8	2.32	0.87	3.87	1/8
Restaurant (Fast Food)	Constant	0	0.26	0.19	1.54	6/8	1.12	0.93	1.26	2/8
		0.5	0.80	0.55	1.39	6/8	1.11	0.93	1.24	2/8
	Occupancy	0	0.93	0.38	1.59	4/8	1.27	0.96	1.59	2/8
		0.5	0.94	0.72	1.49	4/8	1.22	0.96	1.47	2/8
Warehouse	Constant	0	1.03	0.44	1.54	3/8	2.32	1.10	4.18	0/8
		0.5	1.34	0.78	1.66	3/8	2.56	1.20	4.00	0/8
	Occupancy	0	0.92	0.59	1.69	5/8	1.88	0.97	4.38	1/8
		0.5	1.01	0.71	1.73	4/8	2.04	0.88	3.97	1/8
Single-Family House	Constant	0	0.09	0.03	0.18	8/8	3.23	1.87	4.71	0/8
		0.5	0.26	0.06	0.44	8/8	3.63	2.70	4.72	0/8
	Occupancy	0	0.18	0.11	0.31	8/8	1.02	0.60	1.86	4/8
		0.5	0.29	0.16	0.56	8/8	2.48	0.91	3.52	1/8



(a) Small Office (b) Medium Office (c) Fastfood

(d) House (e) Warehouse (f) Retail

Figure 7: Per step reward distribution aggregated over the 8 buildings of the small test set for occupancy based setpoints.

at a 5-minute resolution. All models are trained with PPO for 4 million environment steps, using policy and value networks with two hidden layers of 256 neurons and tanh activations. The per-building specialists (1) are each trained on a single test building. The non-specialist (2) and parameterized (3) models are trained on 16 environments running in parallel, each sampling a new building from the training set at every episode reset; both are trained with 9 random seeds. The parameterized model augments the observation with five normalized building parameters: floor area, construction year, number of warm-up phases, number of actuators, and number of units. In Figure 2a, bars report the median normalized return over the 100 test buildings; error bars show the standard error across the 9 per-seed medians for the multi-building models, and a bootstrap standard error across buildings for the specialists.

F.3 Cross-domain transfer

Let $\mathrm{Emb} = \mathbb{R}^n$ denote the embedding space for a fixed embedding dimension n. The learnable components of the policy are (i) type-specific linear encoders $\{e_t : S(t) \to \mathrm{Emb}\}_{t \in T}$, (ii) a multi-layer transformer $\mathrm{trans} : \mathrm{Emb}^{|V|} \to \mathrm{Emb}^{|V|}$, and (iii) type-specific linear decoders $\{d_t : \mathrm{Emb} \to A(t)\}$ for the node types t.

Policy evaluation proceeds as follows:

function POLICY(s_{raw})
  $s_v \leftarrow \mathsf{split}(s_{\text{raw}})_v \quad \forall v$
  $i_v \leftarrow e_{t(v)}(s_v) \quad \forall v$
  $o \leftarrow \mathsf{trans}(i)$
  $a_v \leftarrow d_{t(v)}(o_v) \quad \forall v$
  return $\mathsf{join}(a)$
end function