Graphs for Scalable Building Decarbonisation: A Transferable Approach to HVAC Control

Anaïs Berkes
Department of Computer Science & Technology
University of Cambridge
Cambridge, United Kingdom
amcb6@cam.ac.uk

Donna Vakalis
Mila, Universite de Montreal
Montreal, Canada
donna.vakalis@mila.quebec

David Rolnick
Mila, Universite de Montreal
Montreal, Canada
drolnick@mila.quebec

Yoshua Bengio
Mila, McGill University
Montreal, Canada
yoshua.bengio@mila.quebec

Abstract

Direct building CO_2 emissions need to halve by 2030 to get on track for net zero carbon building stock by 2050 [1]. Buildings consume 40% of global energy, with HVAC systems responsible for up to half of that demand [2]. Limiting global warming to 1.5°C requires immediate deployment of scalable building efficiency solutions. However, current approaches fail to scale [3]. We introduce HVAC-GRACE (Graph Reinforcement Adaptive Control Engine), the first graph-based RL framework for building control that enables zero-shot transfer by modeling buildings as heterogeneous graphs and integrating spatial message passing directly into temporal GRU gates. Our architecture supports zero-shot transfer by learning topology-agnostic functions. Our framework enables scalable, transferable building control that could accelerate global decarbonisation.

1 Introduction

Buildings are a major carbon source. HVAC (heating, ventilation, air-conditioning) drives an estimated 40-60% of those emissions, representing 12-15 % of global energy-related emissions [2]. Meeting our global emissions targets is going to require faster deployment of emissions-saving measures in the HVAC sector. Although advanced control strategies could substantially reduce HVAC energy consumption, most buildings still rely on (inefficient) rule-based thermostats with static setpoints [4].

2 Why Current Methods are Inadequate

While model-based, data-driven, and learning-based methods show promise, they lack generalisation and require extensive training, limiting deployment across buildings [5, 3]. Model Predictive Control requires costly building-specific models, while RL suffers from sample inefficiency, often needing years of training [6] during which buildings experience suboptimal performance [7]. Pretraining in simulation is impractical due to simulator development costs. Furthermore, policies struggle to transfer between buildings due to varying characteristics [8, 9], requiring retraining for each deployment. Recent RL and transfer-learning studies [10, 11] and morphology-aware methods in robotics [12, 13, 14] show that structure helps, but buildings pose unique heterogeneity and coupling challenges.

Tackling Climate Change with Machine Learning: workshop at NeurIPS 2025.

Our work addresses a fundamental limitation: treating buildings as generic control problems using flat policies applied to concatenated state vectors that ignore inherent structural organisation. Buildings exhibit complex spatial and temporal relationships: zones have thermal adjacency determining heat transfer, weather affects zones differently based on orientation, and HVAC equipment has local effects propagating through structure. These relationships remain consistent across conditions, suggesting that structure-aware policies could transfer more effectively than those learning implicit representations from scratch.

Research in robotics demonstrates that structure-aware approaches improve RL performance. NerveNet [12] showed that encoding morphological structure as graphs enables better sample efficiency and generalisation by sharing parameters across components and modeling physical relationships. This trend extends to other morphology-aware methods: SMP [13] leverages graph representations for similar benefits, while Metamorph [14] uses morphology-aware transformers to capture structural dependencies. However, buildings present distinct challenges: heterogeneous node types with different thermal properties, complex multi-timescale dynamics, and variable control topologies requiring specialised architectural considerations.

Contributions: We introduce **HVAC-GRACE** (**G**raph **R**einforcement **A**daptive **C**ontrol **E**ngine), the first transferable graph-based RL framework for scalable building decarbonisation. We contribute (i) heterogeneous graph representations for building thermal physics, (ii) a unified spatial–temporal Graph RNN with zero-shot transfer capabilities, and (iii) deployment criteria targeting maximum climate impact, potentially preventing 165 megatonnes CO_2 annually toward 2030 climate targets.

3 Methodology

3.1 Heterogeneous Building Graph Construction

We represent buildings as heterogeneous directed graphs $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with three node types:

Conditioned zones ($\mathcal{V}_c$): Indoor spaces with thermostat control that receive local observations and generate control actions. **Unconditioned zones** ($\mathcal{V}_u$): Indoor spaces without active temperature control that provide thermal context. **Outdoor environment** ($\mathcal{V}_o$): A single node aggregating weather conditions and temporal features.

Edge relationships $\mathcal{E}$ capture thermal connections: *thermal adjacency* edges connect zones sharing surfaces, *environmental influence* edges connect outdoor nodes to zones with exterior surfaces, and *self-loop* edges enable temporal state maintenance. More details on the problem formulation can be found in Appendix A.

3.2 Temporal-Spatial Policy Architecture

Our policy processes heterogeneous building states through two integrated stages, where spatial and temporal processing are unified within Graph RNN cells.

Stage 1: Input Processing and Heterogeneous Graph Construction Raw observations are processed through type-specific input MLPs: $x^t = \text{InputMLP}(\text{parsed\_obs})$, where observations are structured as node dictionaries with zone temperatures, weather data, and temporal features mapped to their respective node types. This preprocessing is essential because building zones have different thermal characteristics (conditioned vs. unconditioned vs. outdoor) and require type-specific feature encoding to capture their distinct thermal behaviours and control capabilities.

Stage 2: Integrated Spatial-Temporal Processing via Graph RNN Traditional approaches handle temporal and spatial dependencies separately: RNNs capture temporal patterns but ignore spatial relationships [15], while GNNs capture spatial relationships but struggle with long-term temporal dependencies [16]. However, in buildings these dependencies are tightly coupled: how a zone's temperature evolves depends critically on what neighbouring zones are doing.

Our Graph RNN unifies these relationships by replacing each GRU gate with a heterogeneous GNN. Standard GRU gates (reset, update, and new gates) are the core mechanism in RNNs for controlling how temporal memory is updated at each timestep. By implementing these gates as GNNs instead of simple linear transformations, we enable spatial context from neighbouring nodes to influence temporal memory updates.

Each heterogeneous GNN (HeteroGNN) performs type-specific message passing: (1) compute messages using functions $f_{\text{msg}}^{(\text{type}(u),\text{type}(v))}(h_u)$ that encode thermal physics relationships, (2) aggregate messages $\bar{m}_v = \text{AGG}(\{m_{u\to v} : u \in \mathcal{N}(v)\})$, and (3) update node representations via $f_{\text{update}}^{\text{type}(v)}(h_v, \bar{m}_v)$. More details can be found in Appendix A.

Stage 3: Type-Specific Action Generation After processing through the Graph RNN, specialised policy heads generate control actions for conditioned zones only. For each conditioned zone $v \in \mathcal{V}_c$, we output Gaussian action distribution parameters:

$$\mu_v, \log \sigma_v = f_{\text{policy}}^{\text{conditioned}}(h_v^t)$$

Actions are sampled as $a_v \sim \mathcal{N}(\mu_v, \exp(\log \sigma_v))$. Figure 1 shows the framework for the successful implementation of HVAC-GRACE.

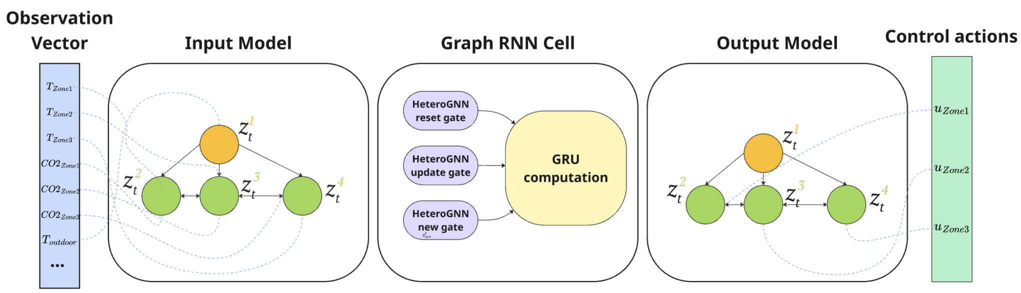

Figure 1: The integration of heterogeneous message-passing GNNs within temporal GRU gates. This spatial-temporal model provides a principled method for learning interactions across building zones.

4 Transferable Architecture and Validation Experiment

HVAC-GRACE enables zero-shot transfer by learning type-specific functions that operate over node types, rather than fixed input positions. Our Graph RNN uses message functions $f_{\text{msg}}^{(\text{type}(u),\text{type}(v))}$, update functions $f_{\text{update}}^{\text{type}(v)}$, and policy heads $f_{\text{policy}}^{\text{conditioned}}$ that generalise across graph structures.

We validated core functionality through zero-shot transfer experiments (Alabama → Montreal climates), confirming successful heterogeneous graph processing. Testing on a Small Hotel model (14% conditioned zones) revealed that sparse connectivity disrupts gradient flow needed for effective spatial reasoning. **Key Insight:** Graph-based control excels in densely connected buildings, which are the highest energy-consuming commercial buildings where climate impact is maximised.

5 Deployment and Climate Impact

Deployment Strategy: Target high-connectivity buildings representing 60% of commercial HVAC consumption. Develop automated topology assessment tools and scale through building management partnerships. **Climate Impact:** Conservative deployment of just 20% of the suitable existing commercial buildings could reduce ~165 000 000 metric tonnes of CO_2 annually – 30% savings in HVAC, across 20% of 50% (commercial only) of the total building stock– which is equivalent to taking 37 million cars off the road [17]. **Economic Viability:** The IEA estimates up to \$2.9 trillion in potential savings from efficient HVAC technologies, creating strong economic incentives for adoption [2]. Zero-shot transfer eliminates retraining costs, topology-agnostic functions enable rapid deployment, and clear ROI metrics drive industry adoption.

6 Acknowledgments

We gratefully acknowledge financial support from BrainBox AI and the Canada CIFAR AI Chairs program, and compute resources from Mila (mila.quebec). We thank colleagues at BrainBox AI for helpful discussions that informed the problem framing; all views and any errors are the authors' alone.

References

[1] Mat Santamouris. Cooling the buildings–past, present and future. *Energy and Buildings*, 128:617–638, 2016.

[2] IEA. Energy efficiency 2024, 2024. Licence: CC BY 4.0.

[3] Zoltan Nagy, Gregor Henze, Sourav Dey, Javier Arroyo, Lieve Helsen, Xiangyu Zhang, Bingqing Chen, Kadir Amasyali, Kuldeep Kurte, Ahmed Zamzam, Helia Zandi, Ján Drgoňa, Matias Quintana, Steven McCullogh, June Young Park, Han Li, Tianzhen Hong, Silvio Brandi, Giuseppe Pinto, Alfonso Capozzoli, Draguna Vrabie, Mario Bergés, Kingsley Nweye, Thibault Marzullo, and Andrey Bernstein. Ten questions concerning reinforcement learning for building energy management. *Building and Environment*, 241:110435, 2023.

[4] Ján Drgoňa, Javier Arroyo, Iago Cupeiro Figueroa, David Blum, Krzysztof Arendt, Donghun Kim, Enric Perarnau Ollé, Juraj Oravec, Michael Wetter, Draguna L Vrabie, et al. All you need to know about model predictive control for buildings. *Annual Reviews in Control*, 50:190–232, 2020.

[5] Zhe Wang and Tianzhen Hong. Reinforcement learning for building controls: The opportunities and challenges. *Applied Energy*, 269:115036, 2020.

[6] Guruvayur Sasikumar Aakash Krishna, Tianyu Zhang, Omid Ardakanian, and Matthew E Taylor. Mitigating an adoption barrier of reinforcement learning-based control strategies in buildings. *Energy and Buildings*, 285:112878, 2023.

[7] Tianyu Zhang, Gaby Baasch, Omid Ardakanian, and Ralph Evins. On the joint control of multiple building systems with reinforcement learning. In *Proceedings of the Twelfth ACM International Conference on Future Energy Systems*, pages 60–72, 2021.

[8] Ilya Zisman, Vladislav Kurenkov, Alexander Nikulin, Viacheslav Sinii, and Sergey Kolesnikov. Emergence of in-context reinforcement learning from noise distillation. *arXiv preprint arXiv:2312.12275*, 2023.

[9] Tianyu Zhang, Mohammad Afshari, Petr Musilek, Matthew E Taylor, and Omid Ardakanian. Diversity for transfer in learning-based control of buildings. In *Proceedings of the Thirteenth ACM International Conference on Future Energy Systems*, pages 556–564, 2022.

[10] Shichao Xu, Yixuan Wang, Yanzhi Wang, Zheng O'Neill, and Qi Zhu. One for many: Transfer learning for building hvac control. In *Proceedings of the 7th ACM international conference on systems for energy-efficient buildings, cities, and transportation*, pages 230–239, 2020.

[11] Anaïs Berkes. Hvac-dpt: A decision pretrained transformer for hvac control. *arXiv preprint arXiv:2411.19746*, 2024.

[12] Tingwu Wang, Renjie Liao, Jimmy Ba, and Sanja Fidler. Nervenet: Learning structured policy with graph neural networks. In *International conference on learning representations*, 2018.

[13] Wenlong Huang, Igor Mordatch, and Deepak Pathak. One policy to control them all: Shared modular policies for agent-agnostic control. In *International Conference on Machine Learning*, pages 4455–4464. PMLR, 2020.

[14] Agrim Gupta, Linxi Fan, Surya Ganguli, and Li Fei-Fei. Metamorph: Learning universal controllers with transformers. *arXiv preprint arXiv:2203.11931*, 2022.

[15] Zachary C Lipton, John Berkowitz, and Charles Elkan. A critical review of recurrent neural networks for sequence learning. *arXiv preprint arXiv:1506.00019*, 2015.

[16] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2009.

[17] EPA. Greenhouse gas emissions from a typical passenger vehicle. Technical report, US Environmental Protection Agency, 2024.

[18] DOE. Energyplus, 2024.

A Appendix

The appendix provides additional technical details supporting the HVAC-GRACE methodology, including the complete problem formulation and training algorithm.

A.1 Problem Formulation

We formulate HVAC control as a Markov Decision Process $(\mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma)$ where: $\mathcal{S}$ includes zone temperatures, outdoor weather (temperature, humidity), time features (hour, day of year), and HVAC energy consumption (electricity, gas); $\mathcal{A}$ represents heating and cooling temperature setpoints for each controllable zone; $\mathcal{P}$ defines physics-based transitions through EnergyPlus [18] simulation; the reward $r(s, a)$ balances energy consumption and comfort violations; and γ is the discount factor.

The agent learns a stochastic policy $\pi_\theta(a_t|s_t)$ that maximises expected discounted return $J(\theta) = \mathbb{E}_{\pi_\theta} \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)$.

A.2 Algorithm

Algorithm 1 presents the complete HVAC-GRACE training procedure, illustrating how heterogeneous building graphs are constructed and processed through our spatial-temporal Graph RNN architecture.

Algorithm 1 HVAC-GRACE Training Algorithm

Input: Building epJSON file, training episodes N
$\mathcal{G} = (\mathcal{V}, \mathcal{E}) \leftarrow$ ConstructGraph(epJSON)
Initialise policy π_θ, critic V_ϕ, Graph RNN states $\{h_v^0\}_{v \in \mathcal{V}}$
for episode $= 1$ to N **do**
  Reset environment, initialise s_0, $\{h_v^0\}_{v \in \mathcal{V}}$
  for timestep $t = 0$ to $T - 1$ **do**
    Stage 1: Input Processing
    parsed_obs $\leftarrow$ ParseObservation(s_t)
    $x^t \leftarrow$ InputMLP(parsed_obs)
    Stage 2: Graph RNN Processing
    $\{h_v^t\}_{v \in \mathcal{V}} \leftarrow$ GraphRNNCell(x^t, $\mathcal{E}$, $\{h_v^{t-1}\}_{v \in \mathcal{V}}$)
    Stage 3: Action Generation
    for each conditioned zone $v \in \mathcal{V}_c$ **do**
      $\mu_v, \log \sigma_v \leftarrow f_{\text{policy}}(h_v^t)$
      Sample $a_v \sim \mathcal{N}(\mu_v, \exp(\log \sigma_v))$
    end for
    Execute actions $\{a_v\}_{v \in \mathcal{V}_c}$, observe s_{t+1}, r_t
  end for
  Update policy π_θ and critic V_ϕ using PPO
end for

We now detail the mathematical operations within Stage 2:

Stage 2: Integrated Spatial-Temporal Processing via Graph RNN

Instead of zones updating memory in isolation, gates perform message passing across the building graph:

$$r_{\text{raw}}^t = \text{HeteroGNN}_{\text{reset}}(x^t, \mathcal{E}) \tag{1}$$

$$z_{\text{raw}}^t = \text{HeteroGNN}_{\text{update}}(x^t, \mathcal{E}) \tag{2}$$

$$n_{\text{raw}}^t = \text{HeteroGNN}_{\text{new}}(x^t, \mathcal{E}) \tag{3}$$

GNN outputs combine with previous hidden states through type-specific transformations:

$$r_v^t = \sigma(r_{\text{raw},v}^t + W_r^{\text{type}(v)} h_v^{t-1}) \tag{4}$$

$$z_v^t = \sigma(z_{\text{raw},v}^t + W_z^{\text{type}(v)} h_v^{t-1}) \tag{5}$$

$$\tilde{h}_v^t = \tanh(n_{\text{raw},v}^t + r_v^t \odot W_h^{\text{type}(v)} h_v^{t-1}) \tag{6}$$

$$h_v^t = (1 - z_v^t) \odot \tilde{h}_v^t + z_v^t \odot h_v^{t-1} \tag{7}$$

This enables spatial context to influence temporal memory updates. When Zone B computes its reset gate while adjacent Zone A is heating, the message function processes Zone A's state, affecting how Zone B updates its thermal memory, anticipating heat transfer and enabling coordinated control decisions.