24th International Conference on Renewable Energies and Power Quality
(ICREPQ'26) Santander (Spain), 10th to 12th June 2026
ISBN-13: 978-84-09-82044-3

Development and HIL Validation of a Low-Cost Two-Level Control Strategy for HVAC Systems

Jesús Clavijo-Camacho[1], Álvaro de la Cruz Álamo[1], Gabriel Gómez-Ruiz[2], Cristian Díaz-Martín[2], Juan P. Torreglosa[1] y Reyes Sánchez-Herrera[1]

[1] Department of Electrical, Thermal, Design and Projects Engineering, University of Huelva
[2] Department of Electronic Engineering, Computer Systems and Automation, University of Huelva

Higher Technical School of Engineering, Campus El Carmen, Avda. de las Fuerzas Armadas, Huelva

Abstract. Buildings account for a substantial fraction of global final energy consumption, with HVAC systems representing their dominant load. The progressive deployment of dynamic electricity pricing creates a strong economic incentive to shift thermal loads toward low-price periods; however, implementing optimization frameworks on low-cost embedded hardware remains largely unvalidated experimentally. This paper presents the design and 24-hour experimental validation of a two-layer hierarchical control architecture for HVAC systems targeted at resource-constrained embedded microcontrollers. The upper layer consists of a rolling-horizon dynamic programming optimizer computing a global binary on/off schedule over a 24-hour planning horizon, while the lower layer implements a real-time safety controller enforcing hard thermal comfort constraints at every control cycle. The architecture is validated through a hardware-in-the-loop platform, in which the microcontroller exchanges signals with a first-order RC building thermal model on a real-time emulation target. Results demonstrate a total electricity cost reduction of 19.71% relative to a conventional hysteresis thermostat, with indoor temperatures maintained within the prescribed comfort band for 98.8% of the trial duration, confirming that cost-optimal thermal management is achievable on severely resource-constrained hardware.

Key words. Hardware-in-the-Loop, HVAC Control, Dynamic Programming, Embedded Microcontroller.

1. Introduction

Buildings are responsible for approximately 40% of total final energy consumption in developed economies, with heating, ventilation, and air conditioning (HVAC) systems accounting for the dominant share of that demand [1], [2]. The progressive deployment of dynamic electricity tariffs — in which the price per kilowatt-hour varies at sub-hourly resolution in response to real-time grid conditions — provides a strong economic incentive for thermal load shifting toward low-price periods. Exploiting this flexibility, however, requires a controller capable of anticipating future price and weather conditions over a sufficiently long planning horizon, rather than reacting solely to instantaneous measurements.

Model predictive control (MPC) and dynamic programming (DP) have been extensively studied as frameworks for cost-optimal building climate management [1] [3]. Both exploit a receding-horizon forecast of outdoor temperature and electricity price to pre-cool or pre-heat the building thermal mass, thereby reducing compressor operation during expensive peak-tariff periods and shifting energy expenditure toward low-cost intervals. Despite a substantial body of simulation-based work demonstrating the economic potential of these strategies, experimental validation on physical or emulated hardware remains comparatively scarce. Deploying predictive controllers in practice introduces challenges absent from simulation-only studies: real-time execution constraints on embedded processors, sensor noise and quantization error, communication latency between system nodes, and the inevitable mismatch between the prediction model assumed by the optimizer and the true plant dynamics. Addressing these challenges rigorously requires a validation methodology that goes beyond software simulation.

Hardware-in-the-loop (HIL) testing provides a controlled and reproducible pathway toward physical validation. In a HIL configuration, the plant — in this context, the thermal dynamics of a conditioned building zone — executes in real time on a dedicated target, while the controller operates on its own hardware and exchanges signals through physical interfaces. This methodology is well established in power electronics [4] and automotive control [5], but its systematic application to building energy management systems remains limited [6]. Critically, the feasibility of executing a full rolling-horizon DP optimization on a severely resource-constrained, low-cost embedded microcontroller [7] — and rigorously validating the closed-loop result through a HIL experiment — has not been demonstrated in the open literature. This represents a meaningful gap between the theoretical maturity of predictive building control and its practical, cost-effective implementation.

This paper makes four original contributions: (C1) a two-layer hierarchical HVAC control architecture combining a rolling-horizon DP optimizer with an independent real-time safety controller; (C2) a complete DP formulation executable on a low-cost embedded microcontroller without floating-point hardware; (C3) a 24-hour closed-loop HIL validation against a 1R1C building thermal model under realistic operating conditions; and (C4) a 19.71% cost reduction over a hysteresis thermostat baseline, with comfort constraints satisfied for 98.8% of the trial. The paper is organized as follows: Section II describes the system architecture and thermal model; Section III details the control formulation; Section IV presents the experimental setup; Section V analyzes the results; and Section VI states the conclusions

2. System Description

A. Overall HIL Architecture

The experimental platform consists of three distinct nodes that interact in closed loops during operation, as illustrated in Fig. 1. The first node is an orchestrator, responsible for constructing the rolling forecast horizon, scheduling periodic re-planning events, and logging experimental data. The second node is an embedded controller, which receives forecast information from the orchestrator, executes the DP optimization algorithm, and issues binary control commands to the plant. The third node is a plant emulator, which emulates the thermal dynamics of the conditioned zone in real time and returns the resulting zone temperature to the embedded controller. The specific hardware instantiation of each node is described in Section IV.

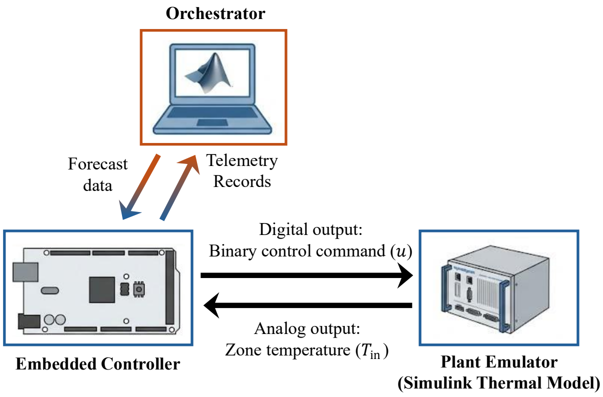

Fig 1. Hardware-in-the-Loop control architecture

B. Thermal Plant Model

The conditioned zone is represented by a lumped first-order thermal network, commonly referred to as the 1R1C equivalent circuit [8] [9]. This modelling abstraction treats the zone as a single homogeneous air mass exchanging heat with the outdoor environment through an equivalent thermal resistance, and subject to active cooling by the HVAC unit. The continuous-time energy balance governing the indoor air temperature is

$$C\frac{dT_{\text{in}}}{dt} = \frac{T_{\text{ext}}-T_{\text{in}}}{R} - u(t)\,P_c \qquad (1)$$

where $T_{\text{in}} \in \mathbb{R}$ is the indoor air temperature (°C), $T_{\text{ext}} \in \mathbb{R}$ is the outdoor air temperature (°C), R(K/W) is the equivalent thermal resistance of the building envelope, C(J/K) is the effective thermal capacitance of the conditioned zone, P_c(W) is the rated cooling power of the HVAC unit, and $u \in \{0,1\}$ is the binary control input. The sign convention is such that $u = 1$ drives T_{in} downward, corresponding to active cooling.

Discretizing (1) using a using a forward-Euler approximation:

$$T_{\text{in}}(k+1) \\ = T_{\text{in}}(k) + T_s\left[\frac{T_{\text{ext}}(k) - T_{\text{in}}(k)}{RC} - \frac{P_c \cdot u(k)}{C}\right] \qquad (2)$$

where $k \in \mathbb{Z}_{\geq 0}$ denotes the discrete time index. This recursion serves a dual purpose in the proposed architecture: it governs the plant emulator executing on the real-time target, and it constitutes the prediction model used by the DP optimizer in the upper control layer. The use of an identical model in both roles is a deliberate design choice that ensures consistency between the cost-to-go values computed during the DP backward pass and the schedule reconstructed during the forward pass, as discussed in Section III.

The 1R1C representation is a well-established approximation for single-zone building thermal analysis [8] [9]. Its principal limitation is the assumption of spatially uniform indoor temperature, which neglects internal heat gains from occupants and equipment, solar radiation through glazing, and multi-zone thermal coupling. These effects introduce a structural mismatch between the model and the true plant dynamics, the consequences of which are analysed in the context of the experimental results in Section V. Numerical values are given in Section IV.

C. Exogenous Inputs

The thermal dynamics described by (1) and (2) are driven by two exogenous input sequences: the outdoor temperature $T_{\text{ext}}(k)$ and the electricity price $\lambda(k)$. Both inputs are assumed to be available as forecast sequences over the planning horizon, denoted $\hat{T}_{\text{ext}}(k)$ and $\hat{\lambda}(k)$ respectively, and both are defined at the discrete time resolution T_s of the plant model. The outdoor temperature forecast constitutes a disturbance input to the thermal dynamics in (2), while the electricity price forecast enters exclusively through the cost function of the DP optimizer, as formalized in Section III. The specific profiles used to drive the experimental validation are described in Section IV.

3. Control Architecture

A. Upper Layer- Rolling Horizon Dynamic Programming

The upper layer solves a finite-horizon optimal control problem at each re-planning instant. Let $k \in \{0,1,\ldots,N-1\}$ denote the discrete time index within the current planning horizon of length N, with each step corresponding to the time interval T_s. The state variable is the indoor temperature $T_{\text{in}}(k)$, the control input is $u(k)$, and the exogenous disturbances are the outdoor temperature forecast $\hat{T}_{\text{ext}}(k)$ and the electricity price forecast $\hat{\lambda}(k)$, both provided by the orchestrator as fixed sequences over the horizon.

1) Problem Formulation. The objective is to minimize the total electricity cost incurred over the planning horizon, subject to the plant dynamics and occupant thermal comfort constraints. The optimal control problem is stated as

$$\min_{u(k)\in\{0,1\}} J = \sum_{k=0}^{N-1} \hat{\lambda}(k) \cdot u(k) \cdot P_c \cdot \frac{T_s}{3600} \quad (3)$$

subject to the plant dynamics given by (2), and the comfort constraint

$$T_{\min} \le T_{\text{in}}(k) \le T_{\max}, \forall\, k \in \{0, \dots, N-1\} \quad (4)$$

where $T_{\min}$ and $T_{\max}$ denote the declared lower and upper comfort bounds respectively. The cost term in (3) evaluates the energy expenditure in monetary units, where the factor $T_s/3600$ converts joules to watt-hours. Rather than applying a continuous relaxation, the problem is solved exactly by dynamic programming over a discretized temperature state grid, as described below.

2) State Discretization. The continuous temperature domain is approximated by a uniform grid of N_Q levels spanning the interval $[T_{g,\min}, T_{g,\max}]$, yielding a grid resolution of

$$\delta T = \frac{T_{g,\max} - T_{g,\min}}{N_Q - 1} \quad (5)$$

The grid boundaries $T_{g,\min}$ and $T_{g,\max}$ are chosen to be sufficiently wider than the comfort band $[T_{\min}, T_{\max}]$ to allow the backward pass to correctly propagate cost-to-go values for states that temporarily violate the comfort constraint during transients, without boundary clipping artifacts.

3) Backward Pass. The value function $V_k(q)$ represents the minimum cost achievable from grid level q at step k through to the end of the horizon. The terminal condition is

$$V_N(q) = 0, \forall\, q \in \{0, \dots, N_Q - 1\} \quad (6)$$

For $k = N - 1$ down to 0, the Bellman recursion is

$$V_k(q) = \min_{u\in\{0,1\}} \{c_k(q, u) + V_{k+1}(q')\} \quad (7)$$

and the stage cost is

$$c_k(q, u) = C_1 + C_2 \quad (8)$$

Where:

$$C_1 = \hat{\lambda}(k) \cdot u \cdot P_c \cdot \frac{T_s}{3600} \quad (8.1)$$

$$C_2 = M \cdot [T_{\text{in}}(k+1) \notin [\tilde{T}_{\min}, \tilde{T}_{\max}]] \quad (8.2)$$

In (8), M is a sufficiently large penalty weight that renders comfort violations non-optimal under any feasible price profile, and $\tilde{T}_{\min}$ and $\tilde{T}_{\max}$ are inner comfort bounds defined as

$$\tilde{T}_{\min} = T_{\min} + \epsilon_{\min}, \tilde{T}_{\max} = T_{\max} - \epsilon_{\max} \quad (9)$$

where $\epsilon_{\min} \ge 0$ and $\epsilon_{\max} \ge 0$ are small tightening margins introduced to create a thermal buffer between the DP operating range and the declared comfort boundaries. This asymmetric tightening serves two purposes: it reduces the frequency with which the lower-layer safety controller must override the DP schedule, and it provides a degree of robustness against the structural model–plant mismatch inherent in the 1R1C approximation. The specific values of M, $\epsilon_{\min}$, and $\epsilon_{\max}$ are reported in Section IV. The optimal policy $\pi_k(q) = \text{argmin}\{\}$ is stored for every (k, q) pair at each backward step.

4) Forward Pass. Once the backward pass is complete, the optimal schedule $\mathbf{u}^* = \{u^*(0), \dots, u^*(N-1)\}$ is reconstructed by simulating the plant model (2) forward from the current measured temperature $\hat{T}_{\text{in}}$, applying at each step the stored optimal policy

$$u^*(k) = \pi_k\left(\text{round}\left[\frac{\hat{T}_{\text{in}}(k) - T_{g,\min}}{\delta T}\right]\right) \quad (10)$$

The forward simulation uses the same discrete model (2), ensuring that the reconstructed schedule is consistent with the cost-to-go values computed during the backward pass.

5) Rolling Horizon Strategy. The DP problem is not solved once at $t = 0$ and executed open loop for the remainder of the horizon. Instead, the orchestrator delivers a fresh N-step horizon window to the embedded controller at regular re-planning intervals of duration T_{upd}, always starting from the current time and extending the full planning horizon into the future. Upon reception of each new forecast frame, the controller re-executes the complete backward–forward DP procedure using the current measured plant temperature $\hat{T}_{\text{in}}$ as the initial condition of the forward pass. Between two consecutive re-planning events, the controller follows the schedule computed at the most recent update, advancing one step per time interval T_s.

The complete DP procedure executed by the embedded controller at each re-planning event is summarized in Algorithm 1.

Algorithm 1. Pseudocode of the DP implemented

```
// Backward pass
VN(q) ← 0, ∀q
for k = N-1 downto 0 do
    for q = 0 to NQ-1 do
        for u ∈ {0, 1} do
          T' ← simulate(Tg(q), T̂ext(k), u)  // eq. (2)
            q' ← clip(round[(T'-Tg,min)/δT], 0, NQ-1)
            cost(u) ← ck(q, u) + Vk+1(q')  // eq. (8)
        Vk(q) ← min{cost(0), cost(1)}
        πk(q) ← argmin{cost(0), cost(1)}
// Forward pass
Tsim ← T̂in
for k = 0 to N-1 do
    qsim ← clip(round[(Tsim-Tg,min)/δT], 0, NQ-1)
    gu(k) ← πk(qsim)
    Tsim ← simulate(Tsim, T̂ext(k), gu(k))
return gu
```

B. Lower Layer- Real-Time Safety Controller

The lower layer operates independently of the DP schedule and executes at every control cycle of duration T_{ctrl}. Its sole function is to enforce the hard comfort constraints (4) in real time by overriding the DP command whenever the measured temperature approaches the declared comfort boundaries. The composite control output of the two-layer architecture is

$$u_{\text{act}}(t) = \begin{cases} 1, & \text{if } T_{\text{in}}(t) > T_{\max} \\ 0, & \text{if } T_{\text{in}}(t) < T_{\min} \\ u^*(k(t)), & \text{otherwise} \end{cases} \quad (11)$$

where $k(t) = \lfloor (t - t_{\text{upd}})/T_s \rfloor$ is the current step index within the active schedule, and t_{upd} is the time of the most recent re-planning event.

The temperature measurement $T_{\text{in}}(t)$ fed to (11) is obtained from the embedded controller's analog input and processed through a noise-rejection filter before use. The raw measurement is smoothed by an exponential moving average (EMA) filter [10]

$$\hat{T}_{\text{in}}(t) = \alpha \cdot T_{\text{raw}}(t) + (1 - \alpha) \cdot \hat{T}_{\text{in}}(t-1) \quad (12)$$

where $\alpha \in (0,1]$ is the smoothing coefficient, and an outlier rejection gate discards any instantaneous reading that deviates from the current filtered estimate by more than a prescribed threshold ΔT_{gate}, replacing it with the previous filtered value. This prevents spurious measurement transients from triggering unnecessary override events. The specific values of α, ΔT_{gate}, and T_{ctrl} adopted in the experimental implementation are reported in Section IV.

4. Experimental Setup

A. Hardware Platform

The orchestrator is a host PC running MATLAB R2024a, executing a fully autonomous polling loop for the 24-hour trial. The embedded controller is an Arduino DUE (Atmel SAM3X8E, ARM Cortex-M3, 84 MHz, 512 KB flash, 96 KB SRAM) [11], operating without a floating-point coprocessor or real-time operating system, with a 12-bit analog input and USB serial communication at 115,200 baud. The plant emulator is a Speedgoat [12] real-time target running a compiled Simulink model at T_{sim} = 0.01 s, outputting zone temperature as a calibrated 0–2.5 V analog signal to pin A0 of the Arduino DUE and receiving the binary control command on a digital input channel. Every T_{upd} = 15 min, the orchestrator transmits forecast data to the embedded controller via a compact binary protocol; upon reception, the controller executes the DP algorithm and returns the optimal schedule for logging. Temperature measurements are averaged over eight consecutive ADC readings, smoothed by the EMA filter (12) with α = 0.15 and outlier gate ΔT_{gate} = 1.0 °C. The safety controller executes at T_{ctrl} = 1 s.

B. Model and Control Parameters

All model, comfort, and DP parameters are listed in Tables I and II. The thermal model corresponds to a light-construction single zone of approximately 20 m². The inner bound tightening margins ($\epsilon_{\min}$ = 0.2 °C, $\epsilon_{\max}$ = 0.5 °C) are set asymmetrically to provide greater margin against upper-bound violations during peak thermal load periods.

TABLE I. Thermal model and comfort parameters.

Parameter	Symbol	Value	Unit
Thermal resistance	R	0.2014	K/W
Thermal capacitance	C	1.208×10^4	J/K
Cooling power	P_c	1500	W
Time step (planning)	T_s	60	s
Comfort lower bound	T_{min}	22.0	°C
Comfort upper bound	T_{max}	26.0	°C

TABLE II. DP algorithm parameters.

Parameter	Symbol	Value	Unit
Horizon length	N	1,440	steps
Planning time step	T_s	60	s
Grid lower bound	$T_{g,\min}$	16	°C
Grid upper bound	$T_{g,\max}$	38	°C
Grid levels	N_Q	50	—
Grid resolution	δT	0.449	°C/level
Re-planning period	T_{upd}	15	min
Penalty weight	M	1,000	—
Lower tightening margin	$\epsilon_{\min}$	0.2	°C
Upper tightening margin	$\epsilon_{\max}$	0.5	°C
EMA coefficient	α	0.15	—
Outlier gate	ΔT_{gate}	1.0	°C
Safety control cycle	T_{ctrl}	1	s

C. Input Profiles

The outdoor temperature follows a warm-climate summer diurnal cycle (peak 36 °C at solar noon, minimum 23 °C). The electricity price profile reflects a time-of-use tariff with five levels (Fig. 2 b)), creating a strong incentive to pre-cool during the cheap midday window (13:00–16:00, 0.02 €/kWh) before the evening peak (16:00–20:00, 0.35 €/kWh). Perfect forecast accuracy is assumed to isolate the effect of the control strategy from forecast uncertainty

Both profiles are defined at 1-minute resolution and stored as $N = 1{,}440$-element vectors, consistent with the planning horizon described in Section III-A. The outdoor temperature profile is used both as a disturbance input to the plant emulator and as the forecast $\hat{T}_{\text{ext}}(k)$ provided to the DP optimizer. In the present experiment, perfect forecast accuracy is assumed — that is, $\hat{T}_{\text{ext}}(k) = T_{\text{ext}}(k)$ for all k— in order to isolate the effect of the control

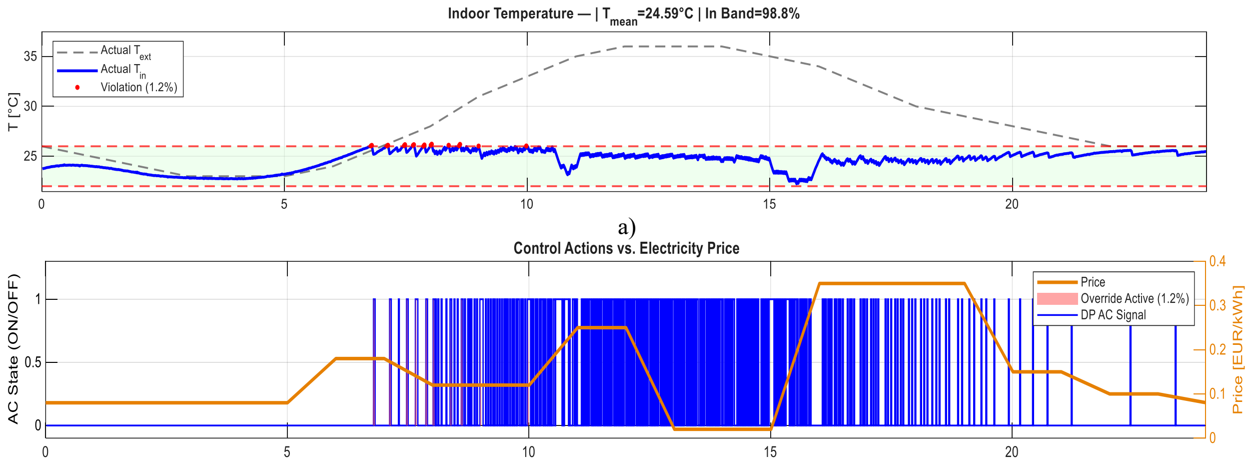

Fig. 2. System behaviour over the 24-hour experimental test. a) Indoor temperature maintained within comfort bounds alongside outdoor temperature variations. b) Dynamic electricity tariff and the binary control command

strategy from forecast uncertainty. The impact of forecast errors is identified as a direction for future work in Section VI

D. Reference Baseline

A hysteresis thermostat is simulated in post-processing over the same 24-hour disturbance sequences, switching cooling on/off at $T_{\min}/T_{\max}$ with no price awareness. Accumulated costs for both controllers are computed via (13) and the relative saving via (14). The baseline is deliberately idealized — no sensor noise or communication delays — so the reported saving constitutes a lower bound on the advantage of the proposed architecture against a real-world thermostat.

$$E(t) = \sum_{i=0}^{k(t)} u(i) \cdot \hat{\lambda}(i) \cdot P_c \cdot \frac{T_s}{3600} \qquad (13)$$

$$\Delta_{\text{cost}} = 100 \cdot \frac{E_{\text{therm}} - E_{\text{DP}}}{E_{\text{therm}}} [\%] \qquad (14)$$

5. Results and Discussion

A. Thermal Performance

The indoor temperature trajectory under the DP controller is shown in Fig. 2, together with the outdoor temperature profile and the comfort band $[T_{\min}, T_{\max}]$. The zone temperature remained within the comfort band for 98.8% of the experiment duration. The only violations recorded were on the upper bound: T_{in} exceeded $T_{\max}$ during 1.2% of samples (~17 minutes), with a maximum of 26.21 °C; no lower-bound violation was observed, consistent with the asymmetric tightening. All upper-bound exceedances were detected and corrected by the safety controller within one control cycle, confirming the correct functioning of the hierarchical architecture

B. Economic Performance

The accumulated cost $E(t)$ is shown in Fig. 3 for both controllers. The DP controller incurred E_{DP}= 1.2026 €, compared with E_{therm}= 1.4977 € for the thermostat, yielding a cost reduction of 19.71% (0.2951 € absolute). The two cost curves diverge most sharply during the evening peak (16:00–20:00, 0.35 €/kWh), when the thermostat continues cycling at the highest tariff while the DP controller, having pre-cooled during the cheap midday window (13:00–16:00, 0.02 €/kWh), keeps the unit off and allows the thermal mass to absorb the load passively. All performance metrics are summarized in Table III.

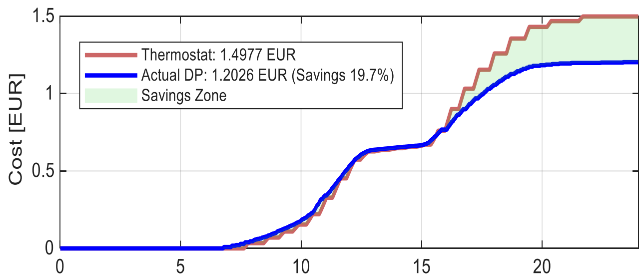

Fig. 3. Comparison of the total accumulated electricity cost between the standard hysteresis thermostat and the proposed Dynamic Programming optimal control approach

TABLE III. Summary of experimental results.

Metric	DP Controller	Thermostat
Total cost (€)	1.2026	1.4977
Absolute saving (€)	0.2951	—
Relative saving (%)	19.71	—
Time within comfort band (%)	98.8	—
Upper-bound violations (%)	1.2	—
Lower-bound violations (%)	0.0	—

C. Discussion

The experimental results validate all four contributions. The two-layer architecture operated correctly throughout the full trial: the DP optimizer generated cost-optimal schedules exploiting the tariff structure — achieving a 19.71% cost reduction consistent with prior simulation-based studies and extending those findings to a hardware-validated setting — while the safety controller enforced comfort constraints engaging only when strictly necessary (1.2% of operating time). The DP backward–forward pass over a 1,440-step horizon executed on the Arduino DUE within a time budget negligible relative to the 15-minute re-planning interval, confirming that globally optimal scheduling is feasible on severely resource-constrained hardware. The 1.2% upper-bound violations were confined to the early-evening transition where the 1R1C model slightly underestimates heat ingress; the maximum exceedance of 0.21 °C above T_max is unlikely to be perceptible to occupants.

6. Conclusion

This paper presented the design and 24-hour HIL validation of a two-layer hierarchical control architecture for cost-optimal HVAC management on a resource-constrained embedded microcontroller. The framework decouples global economic optimization, handled by a rolling-horizon DP algorithm, from real-time comfort enforcement, delegated to an independent safety controller. Experimental results confirm a 19.71% electricity cost reduction relative to a conventional hysteresis thermostat, achieved by concentrating cooling effort during low-price periods and suppressing compressor operation at peak-tariff intervals. Indoor temperatures remained within the comfort band for 98.8% of the trial, with minor upper-bound exceedances confined to 1.2% of operating time and corrected by the safety layer within a single control cycle.

Future work will extend the proposed architecture along four directions: incorporating stochastic formulations to handle forecast uncertainty in outdoor temperature and electricity price; expanding the thermal model to multi-zone representations to assess scalability in larger buildings; integrating the framework with local renewable generation and battery storage within an IoT-enabled building management system; and embedding occupancy prediction models into the upper-layer optimizer to enable demand-responsive comfort scheduling.

References

[1] S. Prívara, J. Široký, L. Ferkl, and J. Cigler, ‘Model predictive control of a building heating system: The first experience’, *Energy Build.*, vol. 43, no. 2, pp. 564–572, Feb. 2011, doi: 10.1016/j.enbuild.2010.10.022.

[2] L. Pérez-Lombard, J. Ortiz, and C. Pout, ‘A review on buildings energy consumption information’, *Energy Build.*, vol. 40, no. 3, pp. 394–398, Jan. 2008, doi: 10.1016/j.enbuild.2007.03.007.

[3] Y. Sun *et al.*, ‘Optimal control strategy for building HVAC systems: Satisfying flexible demand response with different value-based selection’, *Energy Build.*, vol. 323, p. 114823, Nov. 2024, doi: 10.1016/j.enbuild.2024.114823.

[4] R. M. Kennel, T. Boller, and J. Holtz, ‘Hardware-in-the-Loop Systems with Power Electronics: A Powerful Simulation Tool’, in *Power Electronics for Renewable Energy Systems, Transportation and Industrial Applications*, John Wiley & Sons, Ltd, 2014, pp. 573–590. doi: 10.1002/9781118755525.ch18a.

[5] R. Isermann, J. Schaffnit, and S. Sinsel, ‘Hardware-in-the-loop simulation for the design and testing of engine-control systems’, *Control Eng. Pract.*, vol. 7, no. 5, pp. 643–653, May 1999, doi: 10.1016/S0967-0661(98)00205-6.

[6] W. El-Baz, L. Mayerhofer, P. Tzscheutschler, and U. Wagner, ‘Hardware in the Loop Real-Time Simulation for Heating Systems: Model Validation and Dynamics Analysis’, *Energies*, vol. 11, no. 11, Nov. 2018, doi: 10.3390/en11113159.

[7] S. Adhau, S. Dani, D. Ingole, and D. Sonawane, ‘Embedded Model Predictive Controller on Low-Cost Low-End Microcontroller for Electrical Drives’, May 18, 2021, *arXiv*: arXiv:2105.08623. doi: 10.48550/arXiv.2105.08623.

[8] J. Clavijo-Camacho, G. Gomez-Ruiz, J. A. Hernández Torres, and R. Sanchez-Herrera, ‘Day-ahead TCLs dispatch optimization: An integer genetic algorithm approach based on microgrids composition’, *Energy Build.*, vol. 347, p. 116403, Nov. 2025, doi: 10.1016/j.enbuild.2025.116403.

[9] P. Bacher and H. Madsen, ‘Identifying suitable models for the heat dynamics of buildings’, *Energy Build.*, vol. 43, no. 7, pp. 1511–1522, Jul. 2011, doi: 10.1016/j.enbuild.2011.02.005.

[10] A.-V. Oppenheim, A.-S. Willsky, and S.-H. Nawab, *Signals & Systems.: 2nd edition*. Upper Saddle River, NJ, 1997.

[11] ‘Due | Arduino Documentation’. Accessed: Apr. 13, 2026. [Online]. Available: https://docs.arduino.cc/hardware/due/

[12] ‘Speedgoat - Real-Time Simulation and Testing’. Accessed: Apr. 13, 2026. [Online]. Available: https://www.speedgoat.com/