Adaptive Home Heating Control using Recursive Least Squares and Model Predictive Control

Anhar Al Haydar, Emīls Dzintars, Jari de Keijzer, Piotr Kranendonk, Pranav Pisupati, Robin Kruijf
Delft University of Technology
Netherlands

1 Introduction

Residential buildings consume a significant portion of total energy use, with heating systems representing one of the largest contributors [16]. Many domestic heating systems rely on simple thermostatic controllers that activate heating whenever the indoor temperature drops below a predefined threshold following a set schedule. Although this approach is easy to implement, it does not account for future weather conditions, building dynamics, or the carbon intensity of electricity generation of the area or country these houses reside in.

With increasing attention on sustainable software and energy-aware control, there is growing interest in intelligent systems [8] that can shift energy consumption to more favourable periods when energy consumption is low or a large portion of the generated energy is generated sustainably with for instance solar or wind power. In the context of household heating, this means not only maintaining comfort, but also reducing the environmental impact of unwitting heating decisions.

In this project an adaptive heating control strategy is investigated that combines online learning with predictive optimization, in other words a carbon aware thermostat. The proposed system learns the thermal behaviour of a building using Recursive Least Squares and decides how to act in the environment using a Model Predictive Control framework, that is based on predicted weather data and available carbon metrics of energy systems. This model is used to determine heating actions that reduce CO_2 impact while preserving indoor comfort in terms of preset heating schedules.

2 Related Work

Energy-efficient control of building systems has been widely studied in smart energy systems and sustainable computing. Model Predictive Control [7] is a common approach in heating and ventilation optimisation because it allows future system behaviour to be predicted over a finite horizon and enables control decisions to be optimised accordingly.

However, MPC depends on the availability of a sufficiently accurate model of the building. This is challenging in practice because buildings differ in insulation, thermal mass, occupancy patterns, and heating system efficiency.

System identification methods such as Recursive Least Squares [6] offer a practical solution by learning model parameters directly from measurements. By combining RLS with MPC, the controller can continuously improve its representation of the building and adapt its heating strategy over time.

Traditional smart thermostats only optimized for the price consumer would pay, so it only looked at the price per kWh, outside temperature and future predictions.

As global warming becomes more visible, minimizing green house production/consumption becomes more prominent. Heating/cooling uses a significant amount of energy/kWh and so there was a demand to change existing software to try to minimize carbon emissions as well (often with different carbon intensity api's and predictions).

There are already existing solutions, which try to solve the problem. One of them being WattShift [15], WattShift is commercial software, which tries to optimize heating/cooling to minimize costs and carbon footprint. They use electric grid prices, solar/wind production, CO_2 intensity, comfort constraints, etc. As they are a company, they keep their model private with their research.

Another similar solution is Carbon Control [2], which looks at carbon consumption in a broader sense, where it audits a building, office, etc to find a room for improvement, recommend best practices, etc. They must have considered carbon intensity on the grid for heating/cooling as it is where the most energy goes to in a office/building.

3 System Overview

To build a carbon aware thermostat system, both a virtual and a real system had to be designed.

The virtual systems simulates a home setting based on a number of parameters which are discussed in section 3.1. The virtual system implements the temperature dynamics of a 6x10x3 room, and those dynamics are determined by the thermostats heating and the heat transfer between the indoor and outdoor temperature. To properly simulate a full year (2025) of data, only the virtual room system was used to compare the results of the heating algorithm.

The real system predicts the future carbon and temperature intensity values based on previous values gathered from weather and energy data API's. The algorithms are trained on these predictions. The tests with the virtual system however, skip the prediction step and just uses the historical data to train its algorithm.

3.1 Thermal Model

To model the dynamics between indoor temperature, outdoor temperature and heating the following thermal model was used:

$$T_{k+1} = aT_k + b\dot{Q}_{\text{heat,k}} + cT_k^{out} + d \tag{1}$$

where:

- T_k is the current indoor temperature
- $\dot{Q}_{\text{heat,k}}$ is the heating power
- T_k^{out} is the outdoor temperature
- a, b, c, d are model parameters

In the thermal model parameter a represents the effect of the current temperature on the next temperature. Parameter b represents the effect of the heater on the next temperature. Parameter c

represents the effect of the outside temperature on the next indoor temperature. Parameter d represents the effect of the error made by representing the model only by parameters a, b, c.

3.2 Online Residual Least Squares Learning

To learn the values of the model parameters a, b, c, d Residual Least Squares (RLS) Learning is used[5]. At each time step the algorithm compares the indoor temperature with the predicted indoor temperature e_k. The algorithm updates its parameters based on this difference using RLS. Where K_k is the learning rate and λ the forgetting factor (0.9).

$$\begin{aligned} \theta &= [a, b, c, d]^T & \phi_k &= [T_k, \dot{Q}_{\text{heat,k}}, T_k^{out}, 1]^T \\ e_k &= T_{k+1} - \phi_k^T \theta_k & K_k &= \frac{\dot{Q}_{\text{heat,k}} \phi_k}{\lambda + \phi_k^T \dot{Q}_{\text{heat,k}} \phi_k} \\ \theta_{k+1} &= \theta_k + K_k e_k & \dot{Q}_{\text{heat,k+1}} &= \frac{\dot{Q}_{\text{heat,k}} - K_k \phi_k^T \dot{Q}_{\text{heat,k}}}{\lambda} \end{aligned} \tag{2}$$

3.3 Model Predictive Heating Control

Model Predictive Control (MPC) is used to calculate the optimal heating input for the thermostat [9]. It does this by simulating the model using Equation (1), received from the RLS algorithm. MPC simulates this for a horizon of N time steps. It then tries to minimize the following cost function J, by finding an optimal sequence of control inputs u_{k}:

$$J = \sum_{i=1}^{N} \left(w_T |T_i - T_{target,i}| + w_{CO_2}\, \dot{Q}_{\text{heat,i}}\, CI_i \right) \tag{3}$$

$$\dot{Q}_{\text{heat,k}} = 750 \cdot u_k \tag{4}$$

where:

- T_i is the modelled indoor temperature at step i
- $T_{target,i}$ is the desired indoor temperature at step i
- CI_i is the carbon intensity of electricity at step i
- w_T and w_{CO_2} are weighting factors
- $u_k = \begin{cases} 0, & \text{heating off: 0W} \\ 1, & \text{medium power: 750W} \\ 2, & \text{full power: 1500W} \end{cases}$

In Equation (3) the first term penalizes the difference between target and the modelled indoor temperature. The second term penalizes heating during high Carbon Intensity of the electricity network.

Furthermore, the MPC also enforces the following constraint:

$$T_{target,i} - T_{i+1} \leq 3 \tag{5}$$

This ensures that the actual temperature cannot be more then 3 degrees below the target, which prevents MPC from not heating at all because the carbon intensity is very high. At each time step MPC then calculates the optimal heating sequences and applies only the first control action. The optimization is then repeated at the next time step using updated measurements and forecasts.

3.4 Virtual Room System

Because testing the algorithms in real time is extremely time consuming, a Virtual Room was created to simulate temperature changes of a room with specific dimensions and insulation parameters. For these simulations, it is assumed that the whole room consists of only air and the change in temperature can be calculated with:

$$\Delta T = \frac{\dot{Q}_{\text{total}}}{c_{\text{air}} m_{\text{air}}} dt \tag{6}$$

Where:

- dt [s] - is the time step
- ΔT [K] - is the change in temperature per time step
- $c_{\text{air}} = 1006$ J/(kg·K) - is the isobaric specific heat capacity of air
- $m_{\text{air}} = \rho V$ [kg] - is the mass of air in the room
- $\rho_{\text{air}} = 1.204$ kg/m^3 - is the air density
- V [m^3] - is the room volume

The values for c_{air} and ρ_{air} are determined at 20 degrees Celsius from [11] and [13] respectively. To then be able to calculate the total heat transfer, the Virtual Room assumes only the heating from the thermostat and conduction with the outside through walls, as follows:

$$\dot{Q}_{\text{total}} = \dot{Q}_{\text{heat}} - U_{\text{eff}} A (T - T^{\text{out}}) \tag{7}$$

where:

- A [m^2] - is the wall surface of the walls exposed to outdoor temperatures.
- U_{eff} [W/(m^{2}K)] - is the effective thermal conductance

Where the first term in Equation (7) is the heating from the thermostat and the second term is the conduction. For this conduction, the effective conductance of the outside walls is required. This can be calculated as follows:

$$U_{\text{eff}} = \alpha U_{\text{win}} + (1 - \alpha) \frac{1}{\frac{1}{U_{\text{brick}}} + \frac{1}{U_{\text{insul}}} + \frac{1}{U_{\text{brick}}}} \tag{8}$$

$$U_{\text{brick}} = \frac{k_{\text{brick}}}{L_{\text{brick}}},\ U_{\text{insul}} = \frac{k_{\text{insul}}}{L_{\text{insul}}} \tag{9}$$

Where:

- $\alpha = 0.2$ - is the window-to-wall ratio
- $U_{\text{win}} = 1.5$ W/(m^{2}K) - is the thermal conductance of a double glazed window
- U_{insul} [W/(m^{2}K)] - is the thermal conductance of standard insulation
- U_{brick} [W/(m^{2}K)] - is the thermal conductance of a standard brick
- $k_{\text{ins}} = 0.04$ W/(m·K) - is the insulation thermal conductivity
- $k_{\text{brick}} = 0.75$ W/(m·K) - is the insulation thermal conductivity
- $L_{\text{insul}} = 0.15$ m - is the thickness of the insulation layer
- $L_{\text{brick}} = 0.10$ m - is the thickness of a brick

Most outside walls normally consist of both insulated walls and windows. Therefore Equation (8) takes the weighted average of the conductances of both types in the form of the Window-to-Wall ratio (WWR) α. The standard limit of α is 30-40% [1], and is usually around 20% for normal households.

The conductance of a standard window was chosen as a double glazed window from [12]. For the insulation and bricks there is no standard conductance value as these can have varying width and should therefore be calculated from their thermal conductivity with Equation (9). The walls used for the Virtual Room were modelled as two brick layers with an insulation layer in between. Some common values were chosen for the widths of these layers and the thermal conductivity values were chosen for building bricks and mineral wool insulation from [14]. To then get the conductance of the whole wall, the inverse of the sum of inverses of each layers conductance has to be calculated, as can be seen in the second term of Equation (8).

To now update the temperature of this Virtual Room, one only has to provide the active heating power ($\dot{Q}_{\text{heat}}$), the outside temperature (T^{out}) and the time step (dt). The Virtual Room knows all the static parameters and dimensions, as well as its own indoor temperature and then calculates the next temperature using Equations (6) and (7).

4 Implementation

4.1 MPC

Bringing all this together, three different MPC algorithms were implemented and two different sets of preferred temperature schedules were created (one colder and the other warmer) to simulate a whole year of data. The first implemented algorithm, is the Bang-Bang algorithm, which can be used as a baseline simple algorithm. This algorithm polls every minute if the temperature of the room is below the preferred temperature. If the temperature is below the preferred temperature the algorithm gives a heating signal. If the temperature is above the preferred temperature the algorithm stops heating. The second algorithm that is implemented is the MPC algorithm where the weight of the weather data was set to 100 and the weight of the CO_2 emission to 0. So, this algorithm only optimises on the fluctuating outside temperature during the day. The third algorithm that was implemented also used the MPC algorithm where the weight of the weather data was set to 100 and the weight of the CO_2 emission was set to 6. The later weight was chosen by some hyper-parameter tuning, making sure the difference in CO_2 levels match the difference in temperature. All three algorithms were run, using both schedules with different preferred temperatures. The schedule entailed heating in the morning and evening to either 19 or 21 degrees and lowering the preferred temperature during the night.

4.2 Data Collection

For the data collection of the virtual environment, *Met.no* and the *Dutch energy grid* data were used [10][4]. These 2 API's were used to generate a year of data including the temperature at Delft and the energy grid data at the Netherlands. This is the same API as electricity maps uses under the hood for its live Dutch electricity feed. The Carbon signal from the Dutch energy grid was linearly interpolated to get a minute interval. The Bang-Bang algorithm updates the thermostat every minute. The MPC algorithm updates the thermostat every 15 minutes. The reason for only updating once every 15 minutes was to reduce the amount of MPC predictions our solver had to generate over a horizon of N, and therefore the amount of time required to run the algorithm.

5 Results

For an analysis over the entire year, a two-way ANOVA with log-transformation was used. The log-transformation was necessary to handle higher carbon emission variance in winter months compared to summer.

Through observation of a Q-Q plot and histogram on OLS residuals, it was found that the data might not be perfectly normal (throughout the year). However, Levene's test was used to determine that the ANOVA's assumption of homogeneity of variance is not violated. Therefore, the two-way ANOVA was still used as significance test. Since there are around 30 datapoints per month (per algorithm), it is expected that the ANOVA results are sufficiently robust.

For the warmer preferred temperature, a significant difference ($p = 0.0001$) was found between the carbon emissions of the three algorithms when looking over the entire year (see Table 1). Specifically, from Table 2 it can be observed that there was a significant difference in emission between MPC with carbon awareness (MPCC) and Bang-Bang ($p = 0.017$). Furthermore, MPCC had a mean of 1000.20g carbon, MPC without carbon awareness emitted 1065.44g on average, and Bang-Bang had a mean of 1105.68g carbon emission over the whole year. This translates to over 9.5% reduction in carbon emission between MPCC and Bang-Bang (see Table 3).

Table 1: Two-Way ANOVA Results (Warmer Setting)

	sum_sq	df	F	PR(>F)
C(algorithm)	15.93	2.0	9.50	8.15e-05
C(month)	1397.07	11.0	151.46	3.03e-210
Residual	909.00	1084.0	-	-

Table 2: Multiple Comparison of Means - Tukey HSD (Warmer Setting)

group1	group2	meandiff	p-adj	lower	upper	reject
bang	mpc_0	-0.16	0.31	-0.41	0.09	False
bang	mpc_6	-0.29	0.017	-0.55	-0.04	True
mpc_0	mpc_6	-0.14	0.41	-0.39	0.11	False

Table 3: Carbon Savings (Warmer Setting)

algorithm	mean	std	median	Savings (%)
bang	1105.68	1012.80	746.81	0.00
mpc_0	1065.44	1016.72	707.29	3.64
mpc_6	1000.20	981.98	651.86	9.54

For the colder preferred temperature, an even more significant result was found of $p = 5.46 * 10^{-8}$ (see Table 4). From Table 5, it can be seen that there is a significant difference in emission between MPCC and Bang-Bang ($p = 0.0009$) for this temperature setting as well. Moreover, MPCC had a mean emission of 861.1g, MPC without carbon awareness had 929.5g, and Bang-Bang emitted 962.5g on average. Here, MPCC emitted around 10.5% less carbon than Bang-Bang (see Table 6).

Note that, for both warmer and colder preferred temperature, there was only a significant difference found between MPC with carbon awareness and Bang-Bang (see Table 2 and Table 5).

Significance tests were also run when looking at the interaction term (when the month does influence carbon emission). First, for

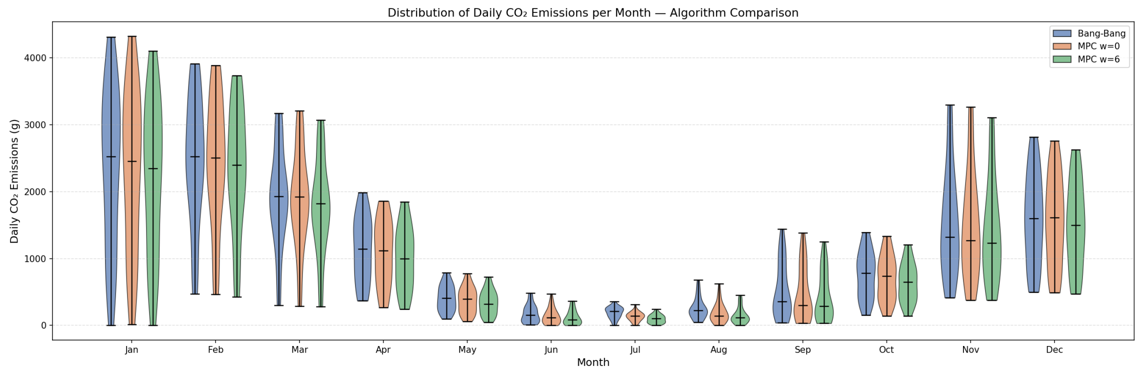

Figure 1: Violin plot for daily co2 emissions (Warm setting)

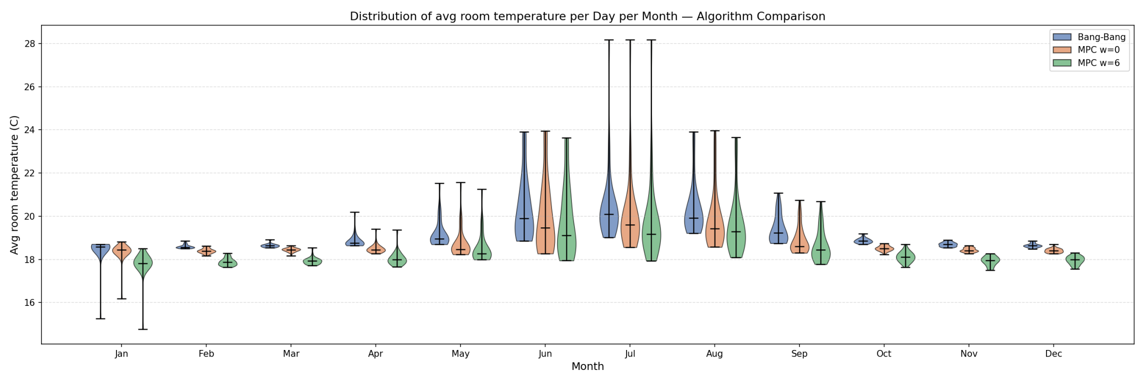

Figure 2: Violin plot for average indoor temperature (Warm setting)

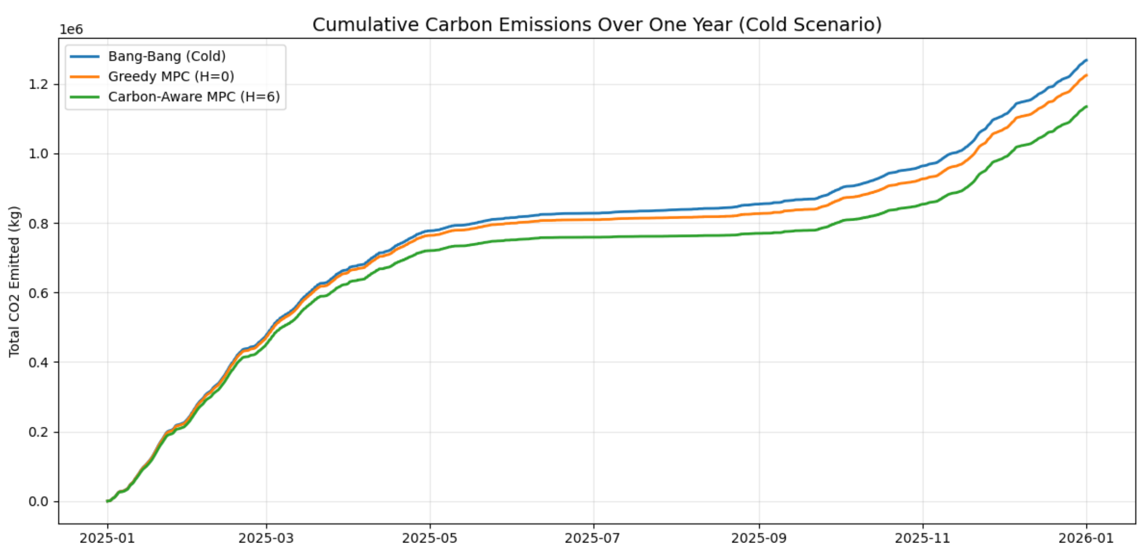

Figure 3: Carbon emissions per year per control strategy

the colder preferred temperature, this interaction term was found to be highly significant with $p = 0.00089$ (see Table 7). Second, for the warmer preferred temperature, it was found that the interaction term was not significant (enough) with $p = 0.507$ (see Table 8), which shows that the carbon-saving benefit of MPC is consistent through the year.

From Figure 1 and from Table 9 it can be observed mpc_6 (also known as MPCC) emitted the least amount of co2, and mpc_0 (MPC without carbon awareness) coming in second. Also, in figure 2, it can be observed that MPC *mpc_6* has the lowest indoor temperature and MPC *mpc_0* is in the second again. To confirm there is also a significant difference between indoor temperatures between different algorithms, statistical tests (Friedman, Wilcoxon with Bonferroni correction) were used, which backed this up for all pairwise combinations of algorithms.

Table 4: Two-Way ANOVA Results (Colder Setting)

	sum_sq	df	F	PR(>F)
C(algorithm)	50.93	2.0	16.98	5.46e-08
C(month)	2640.23	11.0	160.10	3.36e-218
Residual	1625.15	1084.0	-	-

Table 5: Multiple Comparison of Means - Tukey HSD (Colder Setting)

group1	group2	meandiff	p-adj	lower	upper	reject
bang	mpc_0	-0.28	0.14	-0.62	0.07	False
bang	mpc_6	-0.53	0.0009	-0.87	-0.18	True
mpc_0	mpc_6	-0.25	0.20	-0.59	0.09	False

Table 6: Carbon Savings (Colder Setting)

algorithm	mean	std	median	Savings (%)
bang	962.46	954.02	624.08	0.00
mpc_0	929.51	954.31	572.56	3.42
mpc_6	861.10	919.39	509.22	10.53

Table 7: Two-Way ANOVA with Interaction Term (Colder Setting)

	sum_sq	df	F	PR(>F)
Algorithm	50.93	2.0	17.41	3.62e-08
Month	2640.23	11.0	164.13	3.94e-220
Algorithm:month	72.10	22.0	2.24	8.91e-04
Residual	1553.05	1062.0	-	-

Table 8: Two-Way ANOVA with Interaction Term (Warmer Setting)

	sum_sq	df	F	PR(>F)
Algorithm	15.93	2.0	9.49	8.22e-05
Month	1397.07	11.0	151.35	1.34e-208
Algorithm:month	17.81	22.0	0.96	0.507
Residual	891.19	1062.0	-	-

Overall, all three algorithms start out at the same point in terms of total carbon emitted, as seen in Figure 3, but as the year progresses, MPC with carbon awareness stays consistently below the others - especially as he winter months start approaching. The steeper climbs of the other two control types can be attributed to the higher heating demands during quarters 1 and 4. As shown in Figure 4, the mean absolute errors (MAE) which denotes the deviation between target temperature and achieved temperature stays low for the Bang-Bang algorithm but at a very high carbon cost. on the other hand, carbon aware MPC has a temperature deviation of 1.4 degrees but emssions stay low. Notably, the difference in MAE between the standard and carbon aware MPC stays below 1 degree, suggesting that substantial environmental gains can be achieved without compromising perceived indoor comfort.

Table 9: Monthly algorithm rankings by lowest daily energy consumption (warmer and colder preferred temperature)

Month	Algorithm	Winning days (Warm setting)	Winning days (Cold setting)
January	**MPC w = 6**	29/32	30/32
	MPC $w = 0$	2/32	2/32
	Bang-bang	1/32	0/32
February	**MPC w = 6**	26/28	27/28
	MPC $w = 0$	2/28	1/28
	Bang-bang	0/28	0/28
March	**MPC w = 6**	30/31	31/31
	MPC $w = 0$	1/31	0/31
	Bang-bang	0/31	0/31
April	**MPC w = 6**	29/30	29/30
	MPC $w = 0$	1/30	1/30
	Bang-bang	0/30	0/30
May	**MPC w = 6**	30/31	29/31
	MPC $w = 0$	1/31	1/31
	Bang-bang	0/31	1/31
June	**MPC w = 6**	22/30	29/30
	MPC $w = 0$	8/30	1/30
	Bang-bang	0/30	0/30
July	**MPC w = 6**	21/31	31/31
	MPC $w = 0$	10/31	0/31
	Bang-bang	0/31	0/31
August	**MPC w = 6**	22/31	31/31
	MPC $w = 0$	9/31	0/31
	Bang-bang	0/31	0/31
September	**MPC w = 6**	27/30	29/30
	MPC $w = 0$	3/30	1/30
	Bang-bang	0/30	0/30
October	**MPC w = 6**	29/31	30/31
	MPC $w = 0$	2/31	1/31
	Bang-bang	0/31	0/31
November	**MPC w = 6**	29/30	30/30
	MPC $w = 0$	1/30	0/30
	Bang-bang	0/30	0/30
December	**MPC w = 6**	29/31	27/31
	MPC $w = 0$	2/31	4/31
	Bang-bang	0/31	0/31

However, when looked upon carefully, the carbon aware MPC in hot conditions produces a lower MAE as seen in Figure 5, whereas the difference between deviation in temperatures in hot and cold conditions remains marginal for the Bang-Bang algorithm.

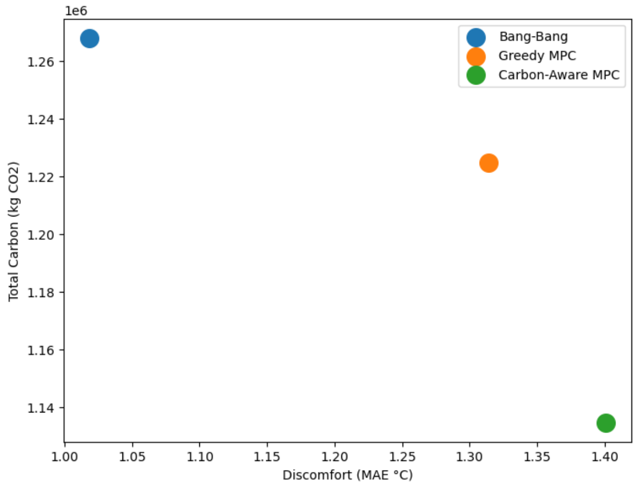

Figure 4: Carbon vs comfort trade-off

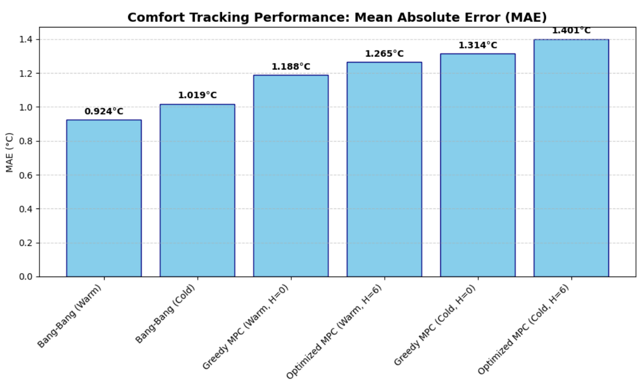

Figure 5: Target temperature deviation per control strategy

6 Discussion

As could be seen in the Results, the carbon-saving benefit of MPC seems to depend on the time of year. It might be that, in some months, the difference in available renewable energy is huge (e.g., windy Spring days). In such months, MPC with carbon awareness seems to have most reduction in carbon emissions in comparison to the baseline. Furthermore, in the middle of winter, heat loss might be fast enough that MPC has less "freedom" to wait for renewable energy, and has to heat immediately to keep the house comfortable. This can explain the significant interaction between emission and month. Moreover, comfort settings (i.e., the preferred temperature) seem to play a big role on MPC's ability to save carbon. For higher preferred temperature, the algorithm appears to save just a small, steady amount every month.

During analysis, it was also found that the indoor temperature was significantly lower for MPC in comparison to the baseline (see Figure 2). There could be a chance that MPCC's emission reduction did not fully come from its intelligent heating ability, but rather from it maintaining a lower indoor temperature.

Since outdoor temperatures peak during summer and can even exceed the indoor target temperature, it is logical that co2 emissions are significantly lower in spring, summer, and autumn compared to winter, as shown in Figure 1. This is because less heating is required during warmer periods. Additionally, Figure 2 shows several outliers in the summer months where temperatures are notably higher than the target temperature. This occurs because, during summer, outdoor temperatures can surpass indoor temperature, leading to occasional overheating.

The testing environment of the runs might not represent the final product correctly. In a previous run the wall thickness was half of what we used in our current setup. Therefore heat dissipated at an extremely fast rate and a model representing a room without walls is an incorrect model. Our current test model could have similar issues and therefore show incorrect or unexpected results.

Using these insights, the algorithms were implemented inside Home Assistant. However, for the final implementation there were numerous issues. The first encountered issue was that the algorithm made use of future data. The algorithm required unavailable predicted data. To overcome the issue of unavailable weather data, predicted data from met.me was used [10]. To overcome the issue of unavailable future carbon intensity data, a rolling window prediction generated by ChatGPT was used to predict values based on history and current carbon intensity data from electricity maps [3]. The second encountered issue, was the incompatibility of using wattage to heat the room. The Home Assistant setup can not change the heating by changing watt. Instead the algorithm maps offset of the thermostat to the u values in the mpc algorithm. A u value of 0w is now mapped to eco-offset, 750w is mapped to the preferred-offset temprature and aggressive-offset maps to the u value of 1500w.

7 Limitations

When designing this system many assumptions were made and trade-off were weighted, resulting in a couple of limitations.

One limitation of the MPC-system is the computational capability. The MPC current algorithms uses 15 minute updates for the thermostat for a more accurate model the thermostat behaviour should be updated every minute. Other limitations in the final product include that most predictions were made using simple algorithms instead of libraries for solving MPC and predicting future carbon intensity values on the grid. The final product is also limited in the cense that it might not resemble the test simulation. For compatibility reasons u values were mapped to offset values in the final product.

8 Conclusion

The study explores an adaptive home heating control strategy using RLS and MPC. The study shows that the carbon aware MPC algorithm performed significantly better compared to the Bang-Bang algorithm, despite updating the thermostat only once per 15 minutes instead of once per minute like Bang-Bang did. On average the proposed method achieves a reduction of carbon emissions of approximately 10%, confirming the effectiveness of the carbon aware algorithm. Further analysis shows that the increased performance of the carbon aware algorithm was more effective during months with higher availability of green energy. During these months the algorithm could better exploit lower carbon energy sources. Furthermore, lower preferred temperatures relatively improved the MPC algorithms. Despite these promising results, the system relies on simplified thermal modals and assumptions about room dynamics, which might not accurately represent a real room. The accuracy of the MPC and RLS algorithms is also highly effected by the forecasts of API tools for the weather and carbon emission. In conclusion, this paper showed promising results for using a carbon aware thermostat, however further work is required to validate the models performance under real world conditions.

References

[1] BEMcyclopedia. *Window-to-wall ratio parametric studies*. 2024. URL: https://bemcyclopedia.com/wiki/Window-to-wall_ratio_parametric_studies (visited on 03/28/2026).

[2] B. Bui and C. De Villiers. "Carbon emissions management control systems: Field study evidence". In: *Journal of Cleaner Production* 166 (2017), pp. 1283–1294.

[3] Electricity Maps. *Electricity Maps: Live CO2 emissions of electricity consumption*. https://app.electricitymaps.com/. Accessed 2026-04-02. 2026.

[4] Energie Data Services Nederland (NED). *NED API Documentation*. https://api.ned.nl/. Accessed 2026-04-02. 2026.

[5] GeeksforGeeks. *Recursive Least Square Algorithm*. Section: Machine Learning. July 9, 2025. URL: https://www.geeksforgeeks.org/machine-learning/recursive-least-square-algorithm/ (visited on 03/31/2026).

[6] Simon S. Haykin. *Adaptive filter theory*. 5. ed. Always learning. Upper Saddle River, NJ: Pearson, 2014. 889 pp. ISBN: 978-0-13-267145-3.

[7] Eurika Kaiser, J. Nathan Kutz, and Steven L. Brunton. "Sparse identification of nonlinear dynamics for model predictive control in the low-data limit". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 474.2219 (Nov. 2018), p. 20180335. ISSN: 1364-5021, 1471-2946. DOI: 10.1098/rspa.2018.0335. arXiv: 1711.05501[math]. URL: http://arxiv.org/abs/1711.05501 (visited on 03/31/2026).

[8] Apoorv Lal and Fengqi You. "Carbon-aware optimization and control in energy systems". In: *Energy and AI* (2025). DOI: 10.1016/j.egyai.2025.100037.

[9] *Model Predictive Control*. In collab. with Steve Brunton. June 11, 2018. URL: https://www.youtube.com/watch?v=YwodGM2eoy4 (visited on 03/31/2026).

[10] Open-Meteo. *Open-Meteo Historical Weather API Documentation*. https://open-meteo.com/. Accessed 2026-04-02. 2026.

[11] The Engineering Toolbox. *Air Density, Specific Weight, and Thermal Expansion Coefficients at Varying Temperatures and Pressures*. 2003. URL: https://www.engineeringtoolbox.com/air-density-specific-weight-d_600.html (visited on 03/28/2026).

[12] The Engineering Toolbox. *Heat Transmission Loss through Building Elements*. 2004. URL: https://www.engineeringtoolbox.com/heat-loss-transmission-d_748.html (visited on 03/28/2026).

[13] The Engineering Toolbox. *Specific Heat Capacity of Air: Isobaric and Isochoric Heat Capacities at Various Temperatures and Pressures*. 2004. URL: https://www.engineeringtoolbox.com/air-specific-heat-capacity-d_705.html (visited on 03/28/2026).

[14] The Engineering Toolbox. *Thermal Conductivity of Common Materials - Solids, Liquids and Gases*. 2003. URL: https://www.engineeringtoolbox.com/thermal-conductivity-d_429.html (visited on 03/28/2026).

[15] WattShift. *Optimize Your Energy Usage*. 2025. URL: https://wattshift.com/ (visited on 04/01/2026).

[16] Fco Javier Zarco-Soto et al. "Carbon-aware energy management in buildings". In: *Energy Reports* (2024). DOI: 10.1016/j.egyr.2024.08.849.