Neural Ordinary Differential Equations for Simulations of Residences Heated with Electric Baseboards

Gabriel Sabbagh[1], Massimo Cimmino[1], Benoit Delcroix[2]
[1]Polytechnique Montréal, Montreal, Canada
[2]Laboratoire des Technologies de l'Énergie, Hydro-Québec Research Institute, Shawinigan, Canada

Abstract

State of the art control schemes for residential HVAC systems are underutilized due to the small margins of improvement compared to the modelling costs. In this study, Neural Ordinary Differential Equations (NODEs) are presented as an approach to address data-driven thermal modelling of residences. These models combine neural networks and differential equations to create continuous dynamic models. To characterise the usefulness of NODEs as a timeseries machine learning technique for residence modelling, their performance is quantified to describe real measurement data. Results show that it is possible to predict the temperatures over 24-hour prediction horizons with an hourly Mean Absolute Error (MAE) of around 0.2 °C.

Introduction

The building sector worldwide consumes almost 120 EJ of energy each year, or more than 30% of the total energy used by all sectors of the economy (IEA International Energy Agency and International Partnership for Energy Efficiency, 2015). In Canadian homes, more than 1400 PJ of energy, almost exclusively in the form of electricity and natural gas, is consumed annually (Natural resources Canada, 2020). Around 80% of this energy is used for space heating and domestic hot water. Measures, like Model Predictive Control (MPC), that optimize the operation and control of different mechanical aspects of residences, such as heating, air conditioning, ventilation, and lighting, seem promising under these premises.

A robust model is necessary for a successful operation optimization strategy. Physical or "White-Box" residence models have historically been widely used for this purpose (Prívara, et al., 2013). These models are considered by the building modelling community to be both interpretable and reliable when developed with accurate information of the properties of the building and its systems. On the other hand, manually designing such models can be a painstaking and time-consuming task. Data-driven or "Black-Box" models are an emerging alternative that can reduce expert involvement in the modelling procedure. These models are based on little to no physical laws but require the exhaustive availability of measurement data in residences to learn how different quantities evolve over time with regards to each other. The increasing affordability and presence of connected devices like thermostats and electricity meters in homes has led to an unprecedented opportunity to access a rich and diverse source of data for Black-Box modelling (Cetin & O'Neill, 2017).

The operation of heating, ventilation, and air conditioning (HVAC) systems in small buildings, such as residences, is usually left to reactive control schemes due to the relatively low payback potential (Huchuk & Bahiraei, 2022). Despite this, recent studies have put forward the potential of using Artificial Neural Network (ANN) based models for predictive control in residences (Demirezen, Fung, & Deprez, 2020; Khan, Shafiq, & Al-Awami, 2023) and have compared them favourably to other Black-Box models in both residential (Potočnik, Vidrih, Kitanovski, & Govekar, 2019) and institutional (Traboulsi & Knauth, 2022) sectors.

Sensors can only give an approximation of the quantities they are measuring. Measured temperatures from thermostats for example, can include noise and patterns induced by instrument sensitivity and sampling frequency. Hence, it can be easy for a Black-Box model to mistakenly include these biases as characteristics of the natural quantity it is modelling. Neural Ordinary Differential Equations (NODEs) (Chen, Rubanova, Bettencourt, & Duvenaud, 2018) have gained traction in time series prediction tasks for several reasons such as their robustness to irregular sampling and their higher scientific interpretability, partly mitigating the aforementioned issues. By integrating differential equations with ANNs, NODEs are regarded as a continuous counterpart to residual neural networks. NODEs make it possible to express the dynamic behaviour of a system using differential equations, while allowing the parameters of these equations to be learned from the data.

Given the relative novelty in both the wide availability of residential measurement data and the development of NODEs, little research has been conducted regarding NODEs for multi-hour residential temperature predictions. To this end, this paper aims to present a methodology for NODE-based residence model development that can be used for MPC in the future. NODE models developed from real datasets are also evaluated as proof of concept.

Methodology

Data

The data used in this study has been collected from one of the Experimental Houses for Building Energetics (EHBE) shown in Figure 1. It is an unoccupied research residence at the "Laboratoire des Technologies de l'Énergie" (LTE), Hydro-Québec's research facility located in Shawinigan, Québec, Canada. The wall assemblies of the building were chosen to represent a typical light-weight wood framed house in Québec constructed in 2011. The residence is heated with baseboard heaters located under the windows of each room and controlled by individual line-voltage programmable thermostats with pulse width modulation at 15 second cycles. The residence is a 2-storey home with an excavated basement. It has a 60 m^2 footprint that excludes the unheated single detached garage. The floor plans as well as the names of the different thermal zones used in this study are given in Figure 4. More information about the residence can be found in Date (2015).

Figure 1: Hydro-Québec's Experimental House of Building Energetics (EHBE)

The available dataset spans a period of 16 days from March 30th to April 15th, 2019, and includes air temperature [°C] and heating load [Wh] (integrated each 15 minutes) for each of the nine thermal zones heated with a baseboard heater, global horizontal solar irradiance [Wm^{-2}] and outdoor dry bulb air temperature [°C]. The dataset has been previously used to calibrate and evaluate Resistance-Capacitance (RC) thermal models of the residence (Vallianos, Athienitis, & Delcroix, 2022) and can be retrieved from Vallianos (2023). The dataset is split into 10 days of training, 2 days of validation and 2 testing datasets of 2 days with a temporal resolution of 1 hour. The choice of using 2 separate test datasets stems from the nature of the available measurements. As shown in Figure 2, the last 2 measured days are atypical compared to the training and validation dataset and exhibit higher temperatures. Hence, the trained model performance is measured when dealing with both data of familiar form (shaded in blue in Figure 2) and data outside its training range (shaded in green in Figure 2). For training and validation, the data is batched into 24-hour batches with a 1-hour sliding window width. All features in the training, validation and testing datasets have been min-max normalized (between 0 and 1) using the training data. In the rest of the article, the predicted data comprised of the zone temperatures will be referred to by x and the exogenous variables (i.e. the quantities not predicted by the model) such as the heating power, the outside dry bulb temperature and the global horizontal irradiance will be referred to by u.

Model

The main idea behind NODEs is to consider the temporal behaviour of a variable, like the evolution of a feature in a neural network, as an ordinary differential equation. Instead of using a traditional numerical integration method to solve this ordinary differential equation, NODEs use a neural network to approximate the solution. In other words, NODEs allow the neural network to learn how variables

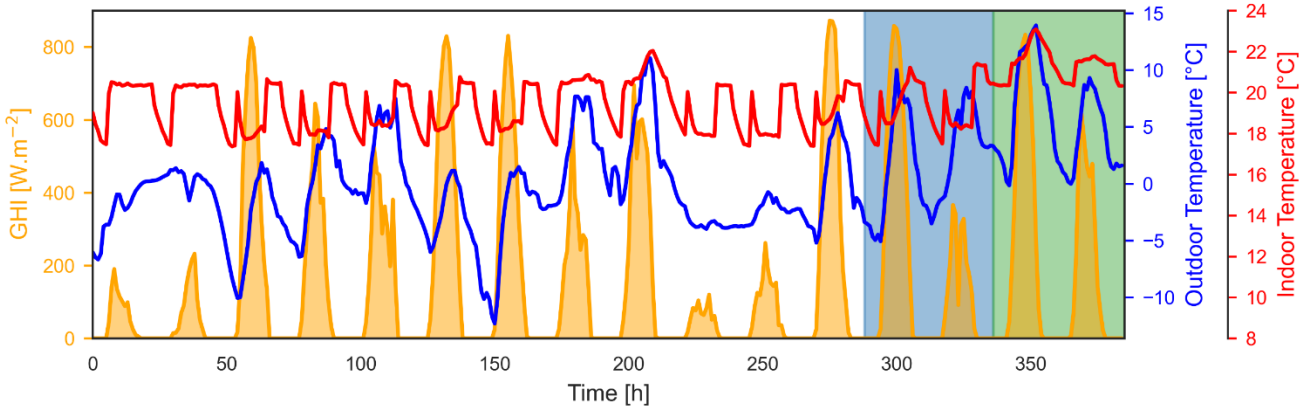

Figure 2: Global Horizontal Irradiance, average residence temperature and outside ambient temperature. Both 2-days testing intervals are shaded in blue and green.

change over time by adjusting its weights and internal parameters.

Let $x(t)$ be the state of a system to model at a time t and let $\frac{dx(t)}{dt} = f(x(t), t, \theta)$ be the derivative with regards to time of the system dynamics with θ as its parameters. Given the state of the system at an initial time $x(t_0)$, it is possible to predict the state of the system at a future time $x(t_1)$ by calculating:

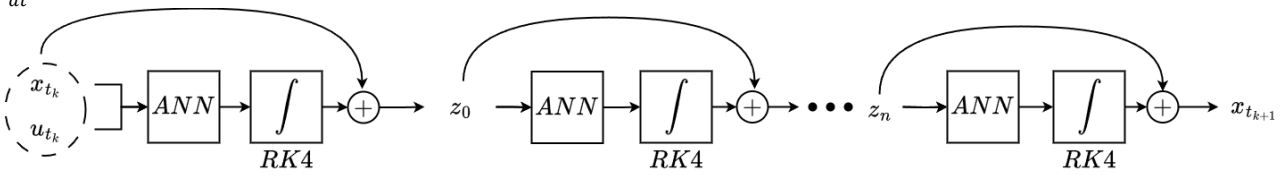

Figure 3: Unrolled NODE architecture.

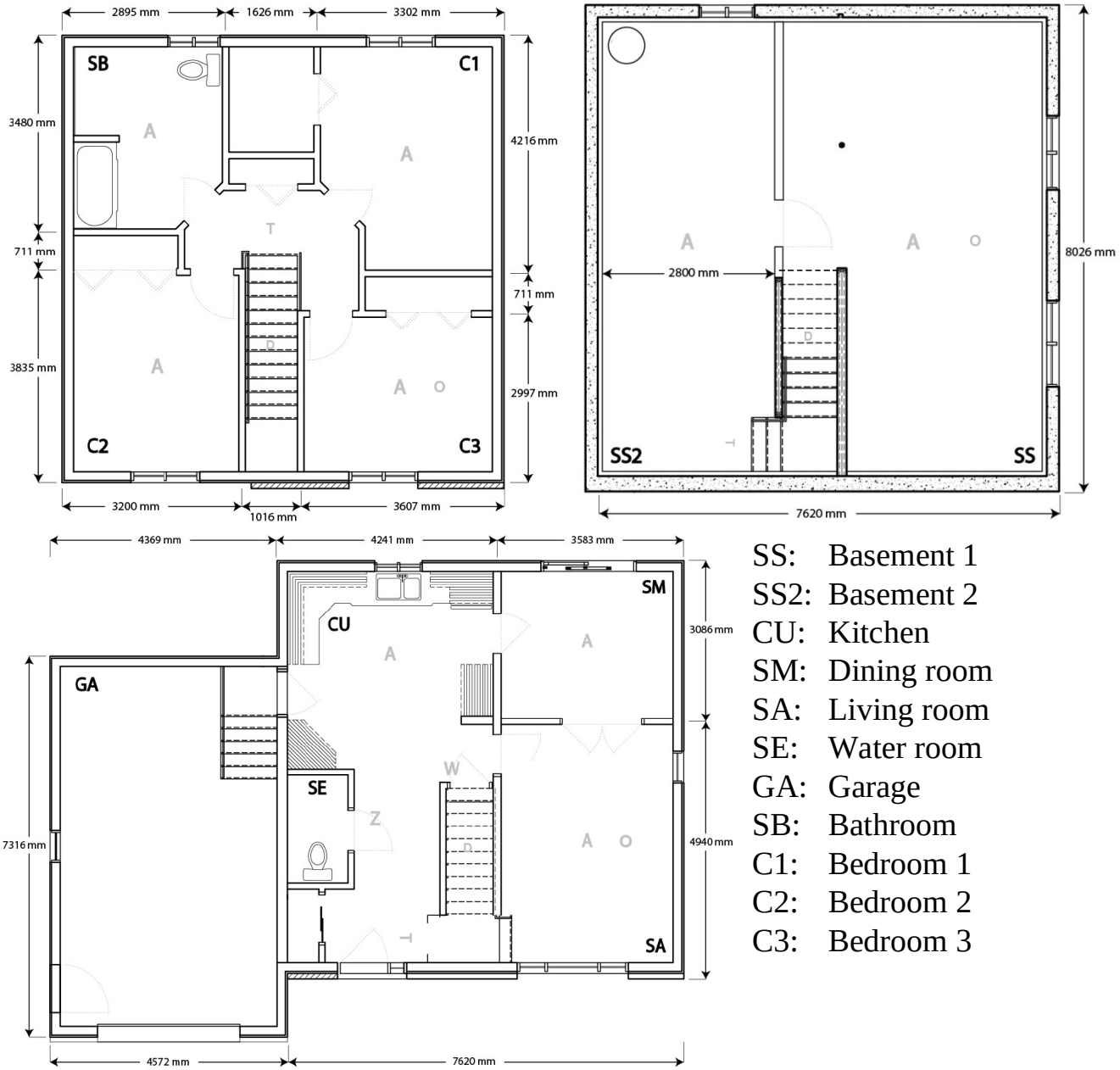

Figure 4: Residence interior schematics. The different letters inside the rooms indicate the locations where temperature measurements are made.

$$x(t_1) = x(t_0) + \int_{t_0}^{t_1} f(x(t), t, \theta)dt \quad (1)$$

and use an ANN to learn the function f. Backpropagating efficiently through a differential equation solver requires the use of the adjoint method (Pontryagin, 1962) when calculating the variation of the loss function with regards to the ANN parameters $\frac{\partial \mathcal{L}}{\partial \theta}$. By letting $a(t) = \frac{\partial \mathcal{L}}{\partial x}$ be the adjoint, $\frac{\partial \mathcal{L}}{\partial \theta}$ is calculated by solving:

$$\frac{\partial \mathcal{L}}{\partial \theta} = -\int_{t_1}^{t_0} a(t)^{\mathrm{T}} \frac{\partial f(x(t), t, \theta)}{\partial \theta} dt \quad (2)$$

A modern mathematical proof can be found in the Appendix of Chen et al. (2018). The model is implemented using NeuroMANCER (Tuor, et al., 2023). The neural network used is a fully connected feedforward ANN with two 24-neuron hidden layers and using a "Softsign" activation function. The integrator is an explicit 4th order Runge-Kutta numerical integrator with an interval of 0.2 hours. Figure 3 illustrates an unrolled representation of the NODE architecture used. On the forward pass of the network, predicted variables x and exogenous variables u measured at a time t_k start by being fed as inputs to the ANN. The ANN produces an approximation of the underlying time dependent dynamics which are then fed to the numerical integrator. The integrator's output is then added to the input of the ANN to generate a state z in the latent space that is subsequently used to generate another latent space state. The procedure is repeated until the arrival to the target vector at time t_{k+1}. This architecture closely mimics the architecture of residual neural networks (He, Zhang, Ren, & Sun, 2015). With NODEs, instead of manually specifying the number of latent states (or in other words hidden layers), their number is defined by the integration sensibility as each intermediate integration step creates an additional latent state. Therefore, the number of layers is determined directly by the integrator and consequently indirectly by the datapoints themselves when using an adaptive step integrator. For this reason, NODEs have been depicted as the "continuous" counterpart of residual neural networks. Figure 5 depicts the architecture of the fully connected ANN used with predicted variables, x, (which are the zone temperatures T in this case) and exogenous variables u, as inputs and $\frac{dx}{dt}$ as outputs. Exogenous variables include each zone's heating input Q, the outdoor dry bulb temperature T_{ext} and the global horizontal solar irradiation GHI. It should be noted that Figure 5 only displays the loss function relevant dynamic outputs which are the variation of the zone temperatures with respect to time $\frac{dT}{dt}$. The loss function used to modify the parameters θ is the loss over the whole trajectory of the predicted integral. This loss can be seen as the point-by-point difference between the predicted trajectory and the actual trajectory of a variable at all the collocation points of a given batch. The loss between a prediction and a real point of index k belonging to the batch number i can be written as:

$$\mathcal{L}_x = \left\| x_k^i - \hat{x}_k^i \right\|_2^2 \quad (3)$$

with x denominating a measurement point and $\hat{x}$ a predicted value. Hence, the objective function to minimize can be expressed as:

$$\min_{\theta} \quad \sum_{i=0}^{M} \sum_{k=0}^{N} \left\| x_k^i - \hat{x}_k^i \right\|_2^2$$
$$s.t. \quad \hat{x}_{k+1}^i = \hat{x}_k^i + \int_{t_k}^{t_{k+1}} f_\theta(\hat{x}_k^i, u_k^i)dt \quad (4)$$

Here, N and M respectively designate the number of colocation points per batch and the total number of batches. u designates exogenous variables fed as entries to the ANN and not used to calculate the loss function. An Adam optimizer (Kingma & Ba, 2015) is used with a learning rate of 0.03. Hyperparameters such as the ANN width, depth, and activation function, integrator interval and optimizer learning rate have been optimized using Optuna (Akiba, Sano, Yanase, Ohta, & Koyama, 2019) on the training and validation datasets. The considered hyperparameter values are presented in Table 1.

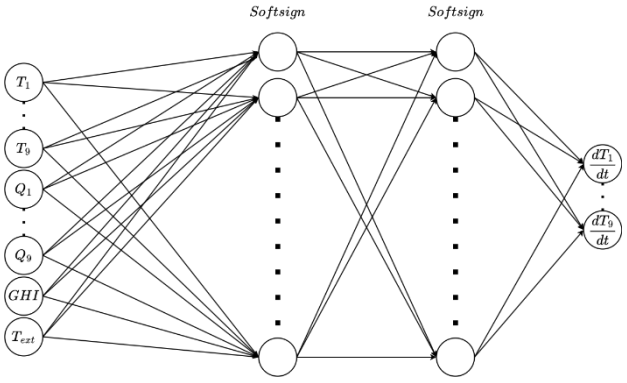

Figure 5: Fully connected ANN used in the proposed NODE architecture.

Table 1: List of optimized hyperparameters.

Hyperparameter	Values considered	Value retained
Number of hidden layers	[1; 2; 3]	2
Neurons per hidden layer	$8n, n \in [\![2,8]\!]$	$n = 3$
Integration interval	$0.1n, n \in [\![1,10]\!]$	$n = 2$
Learning rate	$0.01n, n \in [\![1,10]\!]$	$n = 3$
Activation function	hyperbolic tangent, sigmoid, softsign	softsign

In the context of MPC applications, the model is trained to predict the different thermal zones' temperatures over a 24-hour prediction horizon, hence $N = 24$. During open loop testing, the model is informed of all real measured data (i.e. x_k and u_k) at the start of the 24-hour prediction interval and is only given exogenous variable values u_k at other

indexes k while using the values predicted at index $k-1$ to fill in for the predictions at x_k.

Results

Training and validation datasets

Figure 7 shows the training, validation and testing temperature results for all zones during the 16 days of measured data. The unshaded area corresponds to the training and validation interval. The dashed blue line represents the measured temperature in each zone. At each of the 384 measurement points (240 for training, 48 for validation and 48 for each of the testing sets), the temperature predicted by the model for the next 24 hours is represented with a red curve of reduced opacity. Hence, sections where model predictions coincide, starting from different points in time, result in red curves of higher opacity. The model behaves well over the training interval.

Figure 6 is a boxplot displaying the distribution of the signed difference between the model's 24-hour prediction starting at each measured hour and the measured temperature of each zone over a total of 8940 points (6636 for the combined training and validation intervals and 1152 over each of the testing intervals). A negative difference means an underestimation of the measured temperature by the model. The vertical edges of each box determine the upper and lower quartiles of the error dataset while the vertical line inside the box is the median value. Upper and lower whiskers end at datapoints at or under 1.5 times the interquartile range starting from their respective quartile. Error values that fall outside these whiskers are classified as outliers and are displayed as dots on the plot. The model performs well over the training and validation interval with an overall mean absolute error (MAE) of 0.16°C. Table 2 details the training and validation MAE for the different zones.

Testing datasets

Figure 7 shows the model's predictions over both the 2-day testing intervals, shaded in blue and green. The first testing interval, shaded in blue, corresponds to a dataset resembling the training and validation interval. It can be used to measure the model's performance at predicting zone temperatures under familiar conditions. Figure 6 shows that, within the first testing interval, the model performs slightly worse compared to within the training and validation interval. The model has an overall MAE of 0.35°C for the first testing interval.

The second test interval, shaded in green in Figure 7, corresponds to an atypical dataset when compared to the training and validation interval. The interval features higher peak and average temperatures than the rest of the dataset. This is mainly due, as displayed in Figure 2, to the higher outside ambient temperature combined with the sunny characteristic of both days. The second test interval can be used to measure the model's performance at predicting zone temperatures outside usual conditions. Figure 6 shows that the model performs worse within the second testing interval compared to the first. Figure 6 also displays a predominantly negative signed error between the predicted and the measured zone temperatures within the second testing interval. This is mainly due to the higher average zone temperatures compared to the training and validation interval. The model has a MAE of 0.52°C over the second testing interval. The worsening of the model's performance over the second testing interval puts forward the importance of using an exhaustive training and validation dataset encapsulating as many different temperature states as possible. The zones where the model performs considerably worse within the second testing interval are both basement zones.

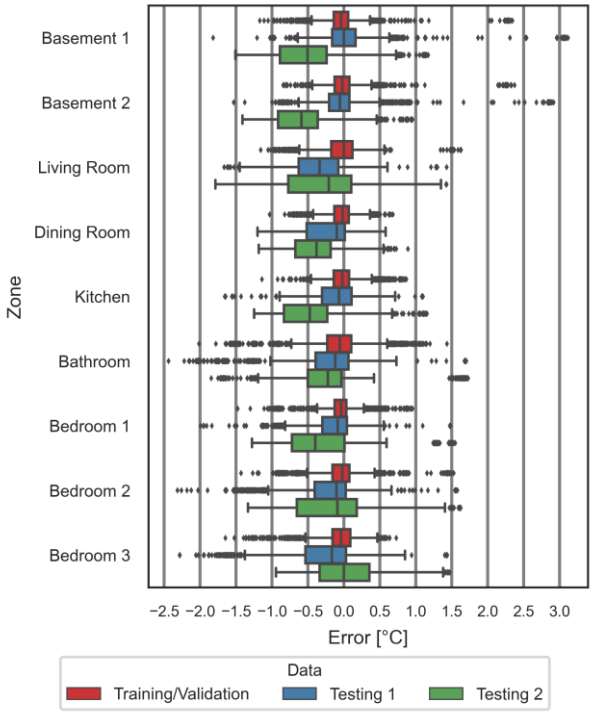

Figure 6: Distribution of the error for each thermal zone for the training and validation dataset as well as both testing datasets.

Table 2: Mean absolute error [°C] of the model temperature prediction over the training and validation and both testing datasets detailed over all the zones.

Zone	Training & Validation	Testing 1	Testing 2
Basement 1	0.15	0.30	0.58
Basement 2	0.14	0.31	0.64
Living room	0.19	0.42	0.57
Dining room	0.13	0.29	0.45
Kitchen	0.14	0.30	0.55
Bathroom	0.23	0.41	0.40
Bedroom 1	0.12	0.29	0.51
Bedroom 2	0.16	0.38	0.57
Bedroom 3	0.16	0.41	0.43

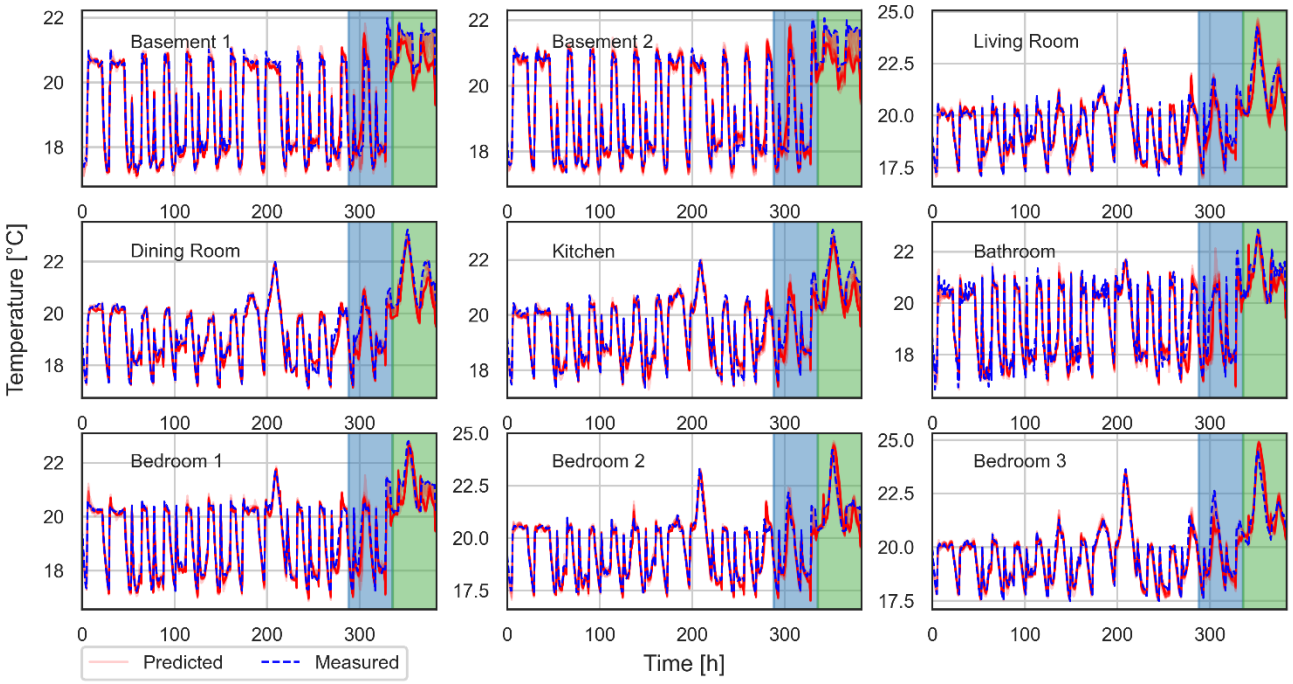

Figure 7: Measured and predicted temperature for each zone. Both 2-days testing intervals are shaded in blue and green. To each measurement point in blue corresponds a 24-hour prediction curve in red with a reduced opacity.

Discussion

Time as an input

In this study, all features provided in the dataset are regularly sampled on an hourly basis. To this end, time is not required to be given as a channel to the model. However, a trained model cannot be expected to retain its accuracy when modelling irregularly sampled data (or partially observed data), as is the case in most thermostats and meters. Even using regularly sampled data with a different sampling frequency from the training data likely requires the reinitialization of the model. NODEs already artificially incorporate a time channel between two observations within the learnt ANN integration scheme; utilizing this aspect of the model to encompass the temporal dependency over several observations is a promising avenue to manage irregular/partially observed data.

Incorporation of prior physical knowledge

As encountered when modelling the second testing dataset, data-driven models usually struggle to preserve their integrity when dealing with conditions outside their training. Even when troubleshooting the cause, and as is the case with all data-driven models, it can be challenging to give physical interpretation to the learnt dynamics. Measuring instruments' resolution and sensitivity can induce noise and unwanted patterns into the measured data and affect the dynamics learned by a model. Incorporating physical knowledge into the model prior to the training phase can help increase its interpretability as well as help steer it more effectively towards favourable behaviour during training. To this end, knowing the physics of how different zones behave thermally, several solutions can be considered such as using more intricate loss functions and ANN architectures.

Computation cost

Both training and usage of the model for simulations were done on a laptop equipped with an AMD Ryzen 7 5800H (3.2 GHz) CPU and a NVIDIA GeForce RTX 3060 Laptop GPU. Training the model was done in under 7 minutes and the 384 simulations plotted in Figure 7 were done in around 1 minute.

Conclusion

To efficiently operate residential heating systems, tailor made predictive models need to be developed for each building. Realistically, these models need to mainly be based on the measured data from the widely deployed thermostats and meters. Using NODEs, an unoccupied residence was modelled using thermostat, meter and weather data over 16 days. The model's performance to predict several zones' temperatures was measured over 24-hour prediction horizons in the context of future MPC use.

While simulating in the conditions of the first testing dataset which features conditions resembling those in the training and validation dataset, the model retains a MAE under 0.5°C for all zones and therefore within the control deadband of most thermostats. This makes it ideal to use as

the model for any advanced heating control scheme. The model's precision degrades while simulating in the conditions of the second dataset featuring conditions outside the training and validation dataset. This puts forward the importance of using an extensive training and validation dataset to calibrate the model and learn the residence thermal dynamics. Future research will aim to make the model more robust while predicting states outside the training conditions.

Additional resources

The code and experimental data used in this paper are available at https://github.com/GabSabb/EHBE-NODE.

Acknowledgment

The authors would like to thank Charalampos Vallianos and Matin Abtahi for making the data used in this study available.

References

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. *The 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, (pp. 2623-2631). Anchorage.

Cetin, K. S., & O'Neill, Z. (2017). Smart meters and smart devices in buildings: a review of recent progress and influence on electricity use and peak demand. *Current Sustainable/Renewable Energy Reports, 4*, 1-7.

Chen, R. T., Rubanova, Y., Bettencourt, J., & Duvenaud, D. (2018). Neural ordinary differential equations. *NIPS'18: Proceedings of the 32nd International Conference on Neural Information Processing Systems*. Montréal.

Date, J. A. (2015). A study of impact of thermal model resolution and zone set point profiles on peak heating load and its calculation, Master's thesis. Concordia University.

Demirezen, G., Fung, A. S., & Deprez, M. (2020). Development and optimization of artificial neural network algorithms for the prediction of building specific local temperature for HVAC control. *International Journal of Energy Research, 44*(11), 8513-8531.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition. *arXiv:1512.03385*.

Huchuk, B., & Bahiraei, F. (2022). Data-driven approaches for predictive thermal modelling in small and medium commercial buildings. *Proceedings of eSim 2022: 12th Conference of IBPSA-Canada*. Ottawa.

IEA International Energy Agency and International Partnership for Energy Efficiency. (2015). *Building energy performance metrics - Supporting energy*. IEA Publications.

Khan, B., Shafiq, S., & Al-Awami, A. T. (2023). Artificial-neural-network-based autonomous demand response controller for thermostatically controlled loads. *IEEE SYSTEMS JOURNAL*, 1-9.

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. *arXiv:1412.6980*.

Natural resources Canada. (2020). *Base de données complète sur la consommation d'énergie*. Retrieved 07 30, 2023, from https://oee.nrcan.gc.ca/organisme/statistiques/bnce/apd/menus/evolution/tableaux_complets/liste.cfm

Pontryagin, L. S. (1962). *Ordinary differential equations*. (L. Kacinskas, & W. B. Counts, Trans.) Addison-Wesley.

Potočnik, P., Vidrih, B., Kitanovski, A., & Govekar, E. (2019). Neural network, ARX, and extreme learning machine models for the short-term prediction of temperature in buildings. *Building Simulation, 12*(6), 1077-1093.

Prívara, S., Cigler, J., Váňa, Z., Oldewurtel, F., Sagerschnig, C., & Žáčeková, E. (2013). Building modeling as a crucial part for building predictive control. *Energy & Buildings, 56*, 8-22.

Traboulsi, S., & Knauth, S. (2022). Machine learning models for predicting indoor air temperature of smart building. *Communications in Computer and Information Science, 1534*, 586-595.

Tuor, A., Drgona, J., Koch, J., Shapiro, M., Vrabie, D., & Briney, S. (2023). *NeuroMANCER: Neural Modules with Adaptive Nonlinear Constraints and Efficient Regularizations*. Retrieved October 03, 2023, from https://github.com/pnnl/neuromancer

Vallianos, C. (2023). *Automated multi-zone model generation (v1.0)*. Retrieved from Zenodo: https://doi.org/10.5281/zenodo.10156745

Vallianos, C., Athienitis, A., & Delcroix, B. (2022). Automatic generation of multi-zone RC models using smart thermostat data from homes. *Energy & Buildings, 277*, 112571.