Data-driven reduced order modelling using clusters of thermal dynamics

Alberto Silvestri[1], Gearóid P. Lydon[2], Christoph Waibel[1], Duan Wu[3], Arno Schlueter[1]
[1]Architecture and Building Systems, ETH Zurich, Zurich, Switzerland
[2]Department of Architecture and Technology, NTNU, Trondheim, Norway
[3]Mitsubishi Electric R&D Centre Europe B.V., Livingston, United Kingdom

Abstract

Physics-based white-box models are employed for estimating building loads and for sizing energy systems at a high level of accuracy. However, these models can become computationally demanding, or might require high technical expertise to be constructed. Therefore, they can become unsuitable for advanced control applications, such as in a model predictive controller (MPC) or for online training of reinforcement learning agents, where simulation speed is crucial. Alternatively, grey-box and black-box models can be employed that require less computational effort while achieving sufficient performance in predicting relevant control signals. Even though black-box models can theoretically represent any dynamical system by implicit inclusion of epistemic unknowns, they lack physical interpretability and require an extensive data set to converge to a suitable model. This work describes the formulation of an ensemble of grey-box models derived from the output of a white-box model from the design phase of a real building. Thus, it presents a novel method to select representative models able to describe the most distinct load curves. As a result, we can reduce the model complexity while maintaining comparable performance at a significantly lower computational cost suitable for control applications.

Highlights

- The study proposes a method for model order reduction to describe a nonlinear system of unknown structure, which does not require expert knowledge to operate.
- A set of most representative second-order grey-box models is generated from the original white-box models.
- The identified grey-box models capture the system dynamics with an error ranging from 0.33 K to 0.51 K, while reducing the computational time by a factor of 100.
- The clustered approach yields better results in terms of RMSE than finding a single general model on the entire data set.
- The proposed method is flexible and can be applied to different white-box models or real systems, with adjustments made to the input vector and RC model structure.

Introduction

Thermal modelling and simulation play a key role in the initial design phase of a building and during its operation. Design models are often formulated knowing the physical properties associated with each building element, and the resulting simulations are used to design and size various systems, including heating, ventilation and air conditioning (HVAC) systems (Farzaneh et al., 2019). On the other hand, the models and simulations used for building operations are essential for predicting future internal thermal conditions, which can be used to optimise the control strategy, such as in the case of model predictive control (Drgoňa et al., 2020) or reinforcement learning (Perera and Kamalaruban, 2021). Both applications require an accurate model of the indoor thermal environment. However, building model development and calibration are often cited as the biggest obstacle to the widespread adoption of model-based control solutions in buildings (Drgoňa et al., 2020). The existing modelling techniques can be divided into three categories: white-box, grey-box and black-box models. They differ based on the data and physical insights included in the model formulation. White-box models are primarily based on physical knowledge. On the opposite side of the spectrum, black-box models rely only on measured data. Between these two extremes, grey-box models combine the information from physical insights and measured data. This approach permits a more straightforward calibration and greater scalability than white-box models, which addresses one of the commonly cited limitations of MPC, i.e. the significant resources, in terms of both time and expert knowledge, required to provide a reliable control model (Drgoňa et al., 2020). One of the reasons why the grey-box approach is often preferred over the black-box is the lack of physical interpretation of the black-box results. The grey-box model has better generalisation capabilities even if the test data is significantly different from the training data (Arendt et al., 2018). White-box models formulated in the building design phase are often too complex to be manipulated without expert knowledge or too heavy for computationally intensive HVAC control approaches, due to computational resource limitations or simulation configuration settings, such as real-time simulation, or when a large number of simulations need to be performed quickly. Model order reduction aims to lower the computational complexity of such problems by reducing the degrees of freedom of the model and therefore approximating the original model. The main qualities of low-order models are a small approximation error compared to the full-order model and the conservation of the properties and dynamic characteristics of the full-order model. In the literature, several model reduction techniques focus on linear systems and assume the full-order model is known (Schilders et al., 2008). The method that we propose is based on identifying low-order resistor-capacitor network (RC) models using the data generated by the full-order model. The novelty of our work lies in the selection of representative models cross-validated with a large ensemble of individual models, thus achieving high generality for distinct load curves. Simulations show that the prediction of the indoor temperature is close to the one generated by the full-order model, with an average root mean square error (RMSE) of less than half a degree kelvin while reducing the computation time by a factor of 100. This study considers the white-box model used in the design phase of the NEST HiLo (High performance - Low emissions) building, which is a recently built research office facility in Zurich, Switzerland (Block et al., 2017). Within HiLo, we focus on *Office 1*, a $22.75\,\mathrm{m}^2$ zone, which faces towards the southwest, as shown in Figure 1. The white-box model of this zone is used to carry out this study.

Figure 1: External view of the HiLo unit from the south-west. Office 1 is located on the HiLo bottom floor.

Methodology

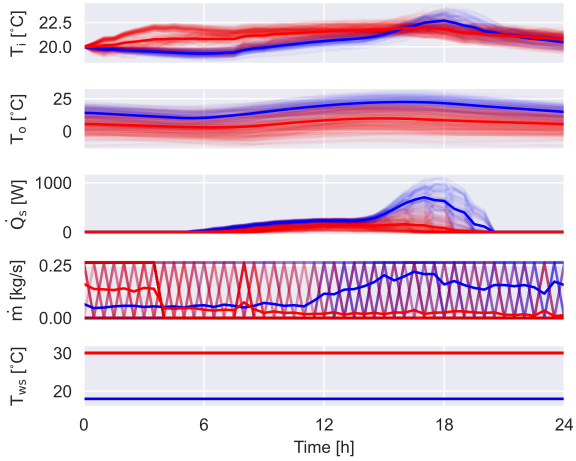

Figure 2: The plot shows the daily overlay of the variables in the data set and their average values. From the top, the subplots represent the room indoor temperature, the outdoor temperature, the solar gains to the room, the water mass flow rate through the TABS, and the TABS water supply temperature. The heating and cooling scenarios are represented in red and blue, respectively.

Before getting to model order reduction, we first describe the full-scale nonlinear model, including its inputs and outputs and the software used. Next, we describe the data set generated by the white-box model, the formulation of reduced order grey-box models from this data, and finally, our approach to identify representative models.

White-box model description

TRNSYS was chosen as the base for the white-box model because of its robustness and broad use in research. TRNSYS is a software environment for simulating transient system behaviour. The program is mostly used in the building domain, although it may also simulate other systems (e.g. traffic flow or biological processes) (Bradley and Kummert, 2005; Duffy et al., 2009).

The nonlinear physics-based TRNSYS model $m_{\mathrm{T}}(\cdot)$ maps the indoor temperature $T_{\mathrm{i},k}$ and the input vector v_k at the current time-step k to the indoor temperature $T_{\mathrm{i},k+1}$ at subsequent time-step $k+1$. The external disturbances of the system, i.e. variables that cannot be controlled, correspond to the vector d_k formed by the outdoor temperature $T_{\mathrm{o},k}$, the solar gains to the zone $\dot{Q}_{\mathrm{s},k}$, the Thermally Activated Building System (TABS) inlet water temperature $T_{\mathrm{ws},k}$. The water mass flow rate of the TABS $\dot{m}_k$ can be controlled by acting on the valve opening u_k. The input vector aggregates the disturbances and the manipulated variable and can be expressed as $v_k = [d_k, \dot{m}_k]' = [T_{\mathrm{o},k}, \dot{Q}_{\mathrm{s},k}, T_{\mathrm{ws},k}, \dot{m}_k]'$. The de-

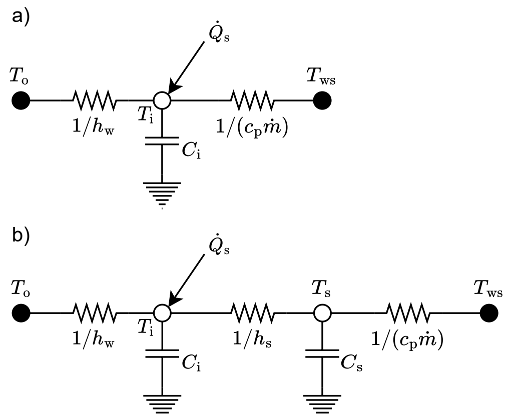

Figure 3: RC representation of the two considered models. a) shows the first-order model, b) shows the second-order model.

scribed relationship can be denoted as:

$$T_{\mathrm{i},k+1} = m_\mathrm{T}(T_{\mathrm{i},k}, v_k, \theta) \tag{1}$$

where θ is the vector collecting the internal parameters of the model. We used this white-box model implementation to generate a data set for the model order reduction process. The original model has been developed and presented in (Lydon et al., 2019).

White-box model generated data set

The TRNSYS model simulated the building systems in *Office 1* for 360 individual days with a time step of 30 min. Each case had varying weather conditions and an optimised valve opening vector u for the TABS. This process generated a data set of 360 independent days (in the following, a day is denoted in a more general sense as a batch) consisting of 48 samples for each measured value. The initial condition for the indoor temperature $T_{\mathrm{i},0}$ is fixed to 20 °C. The TABS inlet water temperature $T_{\mathrm{ws},k}$ is set to 18 °C or 30 °C for the entire duration of the day, according to the cooling or heating scenario, respectively. Figure 2 summarises the data set by showing the overlay of all the batches with the corresponding mean values. On average, during the heating scenario, the solar gains to the room are low. This aspect, combined with a low outdoor temperature, triggers the control system to activate the TABS system on average in the early hours of the day. On the contrary, during the cooling scenario, the outdoor temperatures are higher, as well as the solar gains in the afternoon, being the office oriented towards the southwest. Therefore, the TABS system is utilised more during the second half of the day, on average.

This data set has been generated considering comparable disturbances to those of the real system. However, the RC model parameters identified on this data can be updated when data collected from the real system becomes available, following the same methodology.

Grey-box model formulation

Grey-box models combine previous physical knowledge and statistical knowledge extracted from data. The physical knowledge is encoded in the model by a system of first-order ordinary differential equations (Li et al., 2021).

We use first-order and second-order models in our work, but this workflow can be extended to other model structures or more complex models. The following ordinary differential equation describes the first-order model:

$$C_\mathrm{i}\dot{T}_\mathrm{i} = h_\mathrm{w}(T_\mathrm{o} - T_\mathrm{i}) + \dot{Q}_\mathrm{s} + c_\mathrm{p}\dot{m}(T_\mathrm{ws} - T_\mathrm{i}) \tag{2}$$

where: C_i is the indoor zone thermal capacitance, $\dot{T}_\mathrm{i}$ is the indoor temperature time derivative, h_w is the thermal conductance between the outside air and the indoor temperature, c_p is the specific heat capacity of water at standard conditions. The last term represents the heat gains provided by the TAB system. They have been approximated by using T_i instead of the outlet water temperature. The model has a single state T_i, while the two lumped parameters are grouped in the vector $\theta_1 = [h_\mathrm{w}, C_\mathrm{i}]$.

We extended the model complexity by adding the additional state T_s, which should better represent the overall TABS dynamics. The system of ordinary differential equations then becomes:

$$\begin{cases} C_\mathrm{i}\dot{T}_\mathrm{i} = h_\mathrm{w}(T_\mathrm{o} - T_\mathrm{i}) + \dot{Q}_\mathrm{s} + h_s(T_\mathrm{s} - T_\mathrm{i}) \\ C_\mathrm{s}\dot{T}_\mathrm{s} = h_\mathrm{s}(T_\mathrm{i} - T_\mathrm{s}) + c_\mathrm{p}\dot{m}(T_\mathrm{ws} - T_\mathrm{s}) \end{cases} \tag{3}$$

where: C_i and C_s are the indoor room equivalent thermal capacitance, and the slab equivalent thermal capacitance, respectively, $\dot{T}_\mathrm{i}$ and $\dot{T}_\mathrm{s}$ are the indoor temperature time derivative and slab temperature time derivative, respectively, h_w is the equivalent thermal conductance between the outside air and the indoor temperature, and h_s is the equivalent thermal conductance between the indoor air temperature and the slab temperature. The last term in the second differential equation describes the heat gains provided by the TAB system. They have been approximated by using T_s instead of the outlet water temperature. The four unknown parameters are grouped in the vector $\theta_2 = [h_\mathrm{w}, h_\mathrm{s}, C_\mathrm{i}, C_\mathrm{s}]$.

The models have been discretised and by means of the forward Euler difference method. Therefore, the first-order model can be represented by:

$$T_{\mathrm{i},k+1} = m_1(T_{\mathrm{i},k}, u_k, \theta_1) \tag{4}$$

while the second-order model can be represented by:

$$\begin{bmatrix} T_{\mathrm{i},k+1} \\ T_{\mathrm{s},k+1} \end{bmatrix} = m_2\left(\begin{bmatrix} T_{\mathrm{i},k} \\ T_{\mathrm{s},k} \end{bmatrix}, u_k, \theta_2\right) \tag{5}$$

Grey-box model identification approaches

We used three optimisation approaches to identify the unknown parameter vectors θ_1 and θ_2:

- **Single batch (SB) approach:** The two models have been trained to each of the 360 individual batches by utilising the least square method, which finds the optimal parameters values by minimising the sum of squared residuals J_{SB} of the indoor temperature $T_{\mathrm{i},k}$:

$$J_{\mathrm{SB}} = \sum_{k=1}^{n} \left(T_{\mathrm{i},k} - \hat{T}_{\mathrm{i},k} \right)^2 \tag{6}$$

being n the number of time steps in the batch. The optimisation has been carried out using Python, particularly the *scipy* package's function *least_squares*, which implements the Trust Region Reflective algorithm. This approach generated 360 sets of parameters with different values for each of the two models. However, this approach leads to overfitting for the general case and having a different model for each batch is not in the aim of this work.

- **Whole data set (WD) approach:** Here, we aim to find one model (i.e., the best θ_1 and θ_2 amongst all batches) that can represent all of the 360 days well. We use two alternative methods to increase the chance of finding the global optimum. Firstly, we simply extended the previously described approach. It consists of training the models on a single batch to solve the least square problem, then evaluating them on all the other batches in a k-fold cross-validation fashion (Refaeilzadeh et al., 2016). The model that minimises the cost function J_{WD} is then chosen as the single model representing the data set. J_{WD} describes the average root mean square error over the entire data set:

$$J_{\mathrm{WD}} = \frac{1}{b} \sum_{j=1}^{b} \left(\sqrt{ \frac{1}{n} \sum_{k=1}^{n} \left(T_{\mathrm{i},k} - \hat{T}_{\mathrm{i},k} \right)^2 } \right)_j \tag{7}$$

being n the number of time steps in the batch, and b the number of batches in the data set.

The second method consists in training the models on all the batches at once. This is motivated by the fact that picking the best model using k-fold cross-validation bears to risk of being in a local optimum for the minimisation of Equation 7. Therefore, we used a global optimisation method to find the set of parameters that minimise Equation 7. In particular, we utilised the particle swarm optimisation method (PSO) with the Python package *PySwarms* (Miranda, 2018). We maintained the default hyperparameter values, namely $c_1 = 0.5$, $c_2 = 0.3$, and $w = 0.9$. They represent the cognitive, social and swarm inertia parameters, respectively. The number of particles in the swarm has been set to 100 and the number of iterations limited to $200 * n_\theta$, where n_θ is the number of parameters in the vector θ_1 or θ_2. [1]

Table 1: Identified parameter values.

Model	Parameter	Initial value	Bounds	Value LS	Value PSO
m_1	h_w [W/K]	10	$[1, 10^5]$	1.12	1.93
	C_i [J/K]	4×10^5	$[10^5, 10^{10}]$	1.17×10^8	7.22×10^8
m_2	h_w [W/K]]	10	$[1, 10^5]$	6.73	2.94
	h_s [W/K]	100	$[1, 10^5]$	2.21×10^2	1.58×10^3
	C_i [J/K]	4×10^5	$[10^3, 10^{10}]$	7.01×10^5	4.12×10^6
	C_s [J/K]	5×10^6	$[10^3, 10^{10}]$	8.41×10^7	5.65×10^9

- **Clustered set (CS) approach:** The last method consists of a combination of the two previous approaches. First, we divide the data set into groups of similar dynamics using k-means clustering (MacQueen, 1967). Then, we optimise the model parameters on these clusters using k-fold cross-validation as described in the whole data set approach. This method reduces the number of final models needed to approximate the original TRNSYS model down to four models, while having significantly better results than considering one single model for the whole data set, as shown in the results section. This approach is loosely inspired by (Waibel et al., 2019).

Finally, we compared the average simulation time of the TRNSYS model and the grey-box models. The times reported are the times taken by Python 3.8.5 in running 100 simulations of the zone for 24 hours on a laptop PC with a 2.60GHz Intel Core i7-9850H CPU with six cores and 16 GB of RAM.

Results and discussion

First, the results considering the whole data set are described and then, we focus on the clustered approach.

Whole data set approach

Both set of RC model parameters θ_1 and θ_2 are identified on single batches, minimising the cost function defined by Equation 6. The initial parameter values and boundaries provided to the minimisation algorithm start from educated guesses based on physical values, as shown in Table 1.

The average RMSE on the training batches is 0.86 kelvin for the first-order model and 0.24 kelvin for the second-order model. Based on these results, we consider the second-order model complex enough to capture the dynamics of the TRNSYS model. On the other hand, the first-order model formulation will not be considered for the rest of this work since its performance in this particular case study is inferior to the second-order model.

We ran each model on all the other batches to find a unique model that minimises the overall average

[1] We have also tried CMA-ES by Hansen et al. (2003) using default parameters and the same evaluation budget, but the algorithm could not find satisfactory solutions.

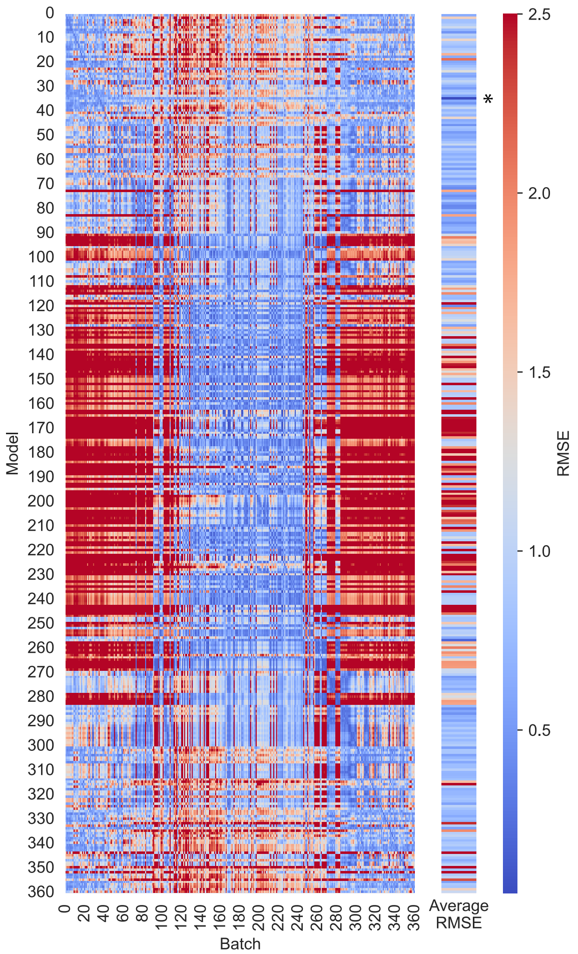

Figure 4: Error heat map representation on the left side. The rows represent the identified models, while the columns represent the batches. The values are the RMSE achieved by the m^{th} model on the b^{th} batch. The average RMSE on all the batches for each m^{th} model is on the right side. The star indicates the model that has the lowest RMSE over the whole dataset.

RMSE on the whole data set. This process resulted in the heat map representation in Figure 4. The rows list the 360 second-order models, while the columns group the batches in the data set. The value of each coordinate is the RMSE achieved by the m^{th} identified model on the b^{th} batch. As expected, the diagonal of the matrix shows the lowest RMSE values, as the models have been identified to the respective batch. In addition, the heat map defines some regions that correspond to different scenarios. A model identified on a specific batch will generally perform well also on batches with similar thermal dynamics. In fact, models that show low RSME values during the summer period (batches 120 to 240) perform worse during winter (batches 0 to 120 and 240 to 360), and vice versa. The 34^{th} model, indicated by the star in Figure 4, minimises Equation 7 with a value of 0.62 kelvin, and it is therefore chosen as the optimal one.

As described in the methodology section, we used global optimisation methods such as PSO to find the best set of model parameters. It found a local minimum with a J_{WD} value of 1.32 kelvin for the first-order model and 1.26 kelvin for the second-order model. The Table 1 reports the optimal parameters for the two approaches.

Although the RMSE is generally low, a closer look at the top plot of Figure 8 shows that this method is not achieving good results in representing the white-box models' dynamics. In particular, this model is not able to capture the indoor temperature peak in the first 12 hours of the day. Two ways are available to enhance the results: one is to increment the model complexity, while the other one is to divide the data set into homogeneous clusters and identify a different model for each cluster. We decided on the latter solution since the current model structure has proven to be complex enough to represent the system dynamics in the training batches.

Clustered set approach

We divided the original data set into clusters with similar indoor temperature dynamics. First, we separated cooling and heating scenarios according to the water supply temperature T_{ws} value, and then we subdivided the heating case. We used the k-means algorithm, an unsupervised learning method. K-means is an iterative algorithm that separates the data set into a pre-defined number of k distinct clusters where each data point belongs to only one group. It assigns data points to a cluster such that the chosen distance metric between the data points and the cluster's centroid is at the minimum (Sinaga and Yang, 2020). The centroids are defined by means of all points that are in the same cluster. The algorithm first chooses random points as centroids and then iterates, adjusting them until full convergence.

Unlike supervised learning, which has a ground truth for assessing model performance, clustering analysis does not have solid metrics that can be used to evaluate the results of various clustering algorithms. In addition, k-means takes the number of clusters k as a hyperparameter. We used the elbow method to get an intuition of how many clusters k are needed. The elbow method computes a curve which shows the $WCSS$ (Within-Cluster Sum of Square) as a function of the number of clusters. $WCSS$ is the sum of the squared euclidean distance between each point and the corresponding cluster's centroid. Figure 5 depicts the elbow curve for the heating scenario data set.

To determine the optimal number of clusters, we selected the value of k at the point after which the $WCSS$ starts to decrease linearly. Thus for the given data, we determined that the optimal number of clusters for the data is 3. We ran the k-means algorithm with this setting, and the results are shown in Figure 6. This procedure divided the heating scenario data

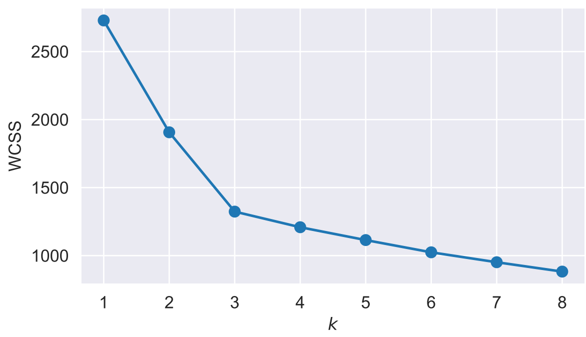

Figure 5: Elbow curve method: it shows the WCSS (Within-Cluster Sum of Square) as a function of the number of clusters k, ranging from 1 to 8. The optimal value of k is determined at the elbow of the curve, i.e. the point where the curve starts to decrease linearly. In this case, the optimal value is $k = 3$.

set into $k = 3$ clusters exhibiting different dynamics, specifically in the first 12 hours. In this period, there could be a pre-heating that is hardly modelled using the whole-data set approach. The three heating clusters, H1, H2, H3, and the cooling case C have 97, 74, 93, and 96 batches, respectively.

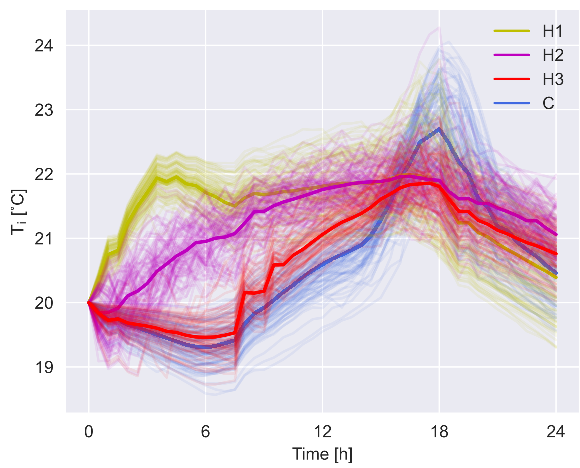

Figure 6: The indoor temperature signals are assigned to the corresponding cluster, whose centroid is represented as a bold line. Each cluster captures a different dynamic behaviour. H1, H2, H3 are the heating clusters identified with k-means, and C is the cooling scenario.

We applied the model identification method described in the methodology section to each of the four clusters. This resulted in four second-order models, having slightly different dynamics according to the respective parameters. Figure 7 shows the identified model parameter value distributions on different clusters. As expected, the parameter variance is reduced, indicating that the optimised parameter models can better capture the system's dynamic considering different batches. The best models of each cluster can outperform the model identified on the whole data set. The average RMSE is 0.42 kelvin for H1, 0.51 kelvin for H2, 0.41 kelvin for H3, 0.33 kelvin for C; considerably better than 0.62 kelvin achieved by the model trained on the whole data set , as shown in Table 2. It is important to notice that the variance in the parameter values in the H2 case are higher than the ones computed for the WD approach mainly because of two factors: the smaller amount of batches available with respect to the other scenarios; and the almost static behaviour of the system in this cluster case, making the optimisation task difficult. Table 2 summarises the final identified parameter values of the second-order RC model for each cluster and for the whole data set approach. One can notice that the thermal capacitance of the TABS C_s in the cooling scenario C is very high and lies at the optimisation boundary. Therefore, the dynamic behaviour of the TABS can be neglected and can be considered as a static component.

Figure 8 gives a qualitative validation of the models in different scenarios. The general model performs worse than the model trained on the specific cluster but on average better than models trained on other clusters since it approximates the average behaviour of the whole data set. Finally, we tested the computation time that the TRNSYS model and the proposed ensemble of grey box models need to simulate a day's worth of data. The TRNSYS model takes, on average, 0.25 second to complete this task, while the model developed with our approach only needs approximately 0.002 second. This comparison confirms that our approach resulted in an extremely lightweight low-order model, which can still approximate the original dynamics with low error.

Table 2: Identified model parameter values and RMSE for each cluster.

Cluster	h_w [W/K]	h_s [W/K]	C_i [J/K]	C_s [J/K]	RMSE [K]
H1	1.47×10^1	6.70×10^1	2.32×10^6	1.14×10^6	0.42
H2	7.22×10^1	5.86×10^2	1.74×10^7	4.59×10^6	0.51
H3	1.14×10^1	7.43×10^1	1.86×10^6	1.50×10^8	0.41
C	1.33×10^1	1.94×10^2	1.88×10^6	9.99×10^9	0.33
WD	6.73	2.21×10^2	7.01×10^5	8.41×10^7	0.62

Conclusions and future work

This study presented a method for model order reduction of a nonlinear system whose model structure is neither known a priori nor requires expert knowledge to be operated. A set of second-order grey-box models identified on clusters of data generated by the white-box model can capture the dynamics with an error ranging from 0.33 K to 0.51 K while reducing the computational time by a factor of 100. The clustered approach yields better results in terms of RMSE than finding a single general model on the whole data set. The proposed identification workflow can be applied to different white-box models or to real systems, adjusting the input vector according to the measured input values, and modifying the RC model structure accordingly. The model performance will be validated

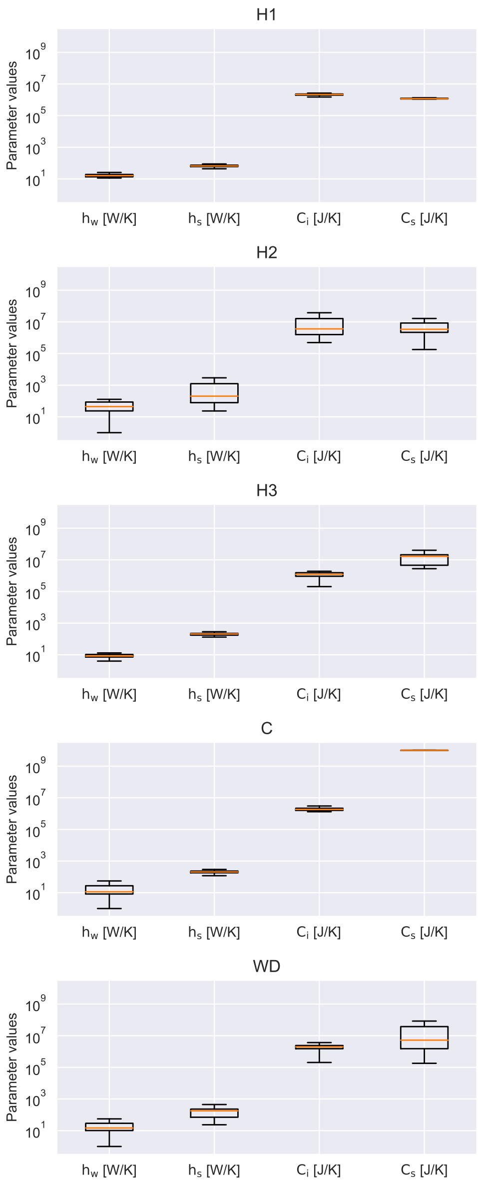

Figure 7: Model parameter box plot. It shows the distribution of the parameter values on different clusters. The y-axis is in logarithmic scale. The outliers are neglected for clarity.

on the data generated by the NEST HiLo unit in future work. Moreover, a technique that determines the cluster case based on the input vector v is under development. Finally, this model will be used to train a reinforcement learning policy that will be deployed to the real building HiLo.

Acknowledgment

This study was partly financed by Mitsubishi Electric R&D Centre Europe B.V. [ETH ID No 16234].

References

Arendt, K., M. Jradi, H. R. Shaker, and C. T. Veje (2018). Comparative Analysis of White-, Gray- and Black-box Models for Thermal Simulation of Indoor Environment: Teaching Building Case Study. pp. 9.

Block, P., A. Schlueter, D. Veenendaal, J. Bakker, M. Begle, I. Hischier, J. Hofer, P. Jayathissa, I. Maxwell, T. M. Echenagucia, Z. Nagy, D. Pigram, B. Svetozarevic, R. Torsing, J. Verbeek, A. Willmann, and G. P. Lydon (2017, July). NEST HiLo: Investigating lightweight construction and adaptive energy systems. *Journal of Building Engineering 12*, 332–341.

Bradley, D. and M. Kummert (2005). New evolutions in trnsys–a selection of version 16 features. In *Proceedings of building simulation*, pp. 107–114. Citeseer.

Drgoňa, J., J. Arroyo, I. Cupeiro Figueroa, D. Blum, K. Arendt, D. Kim, E. P. Ollé, J. Oravec, M. Wetter, D. L. Vrabie, and L. Helsen (2020, September). All you need to know about model predictive control for buildings. *Annual Reviews in Control*, S1367578820300584.

Duffy, M. J., M. Hiller, D. E. Bradley, W. Keilholz, and J. W. Thornton (2009). Trnsys-features and functionalitity for building simulation 2009 conference. In *11th International IBPSA Conference-Building Simulation*, pp. 1950–1954.

Farzaneh, A., D. Monfet, and D. Forgues (2019, May). Review of using Building Information Modeling for building energy modeling during the design process. *Journal of Building Engineering 23*, 127–135.

Hansen, N., S. D. Müller, and P. Koumoutsakos (2003). Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es). *Evolutionary computation 11*(1), 1–18.

Li, Y., Z. O'Neill, L. Zhang, J. Chen, P. Im, and J. DeGraw (2021, August). Grey-box modeling and application for building energy simulations - A critical review. *Renewable and Sustainable Energy Reviews 146*, 111174.

Lydon, G., S. Caranovic, I. Hischier, and A. Schlueter (2019). Coupled simulation of thermally active building systems to support a digital twin. *Energy and Buildings 202*, 109298.

MacQueen, J. (1967). Classification and analysis of multivariate observations. In *5th Berkeley Symp. Math. Statist. Probability*, pp. 281–297. University of California Los Angeles LA USA.

Miranda, L. J. (2018). Pyswarms: a research toolkit for particle swarm optimization in python. *Journal of Open Source Software 3*(21), 433.

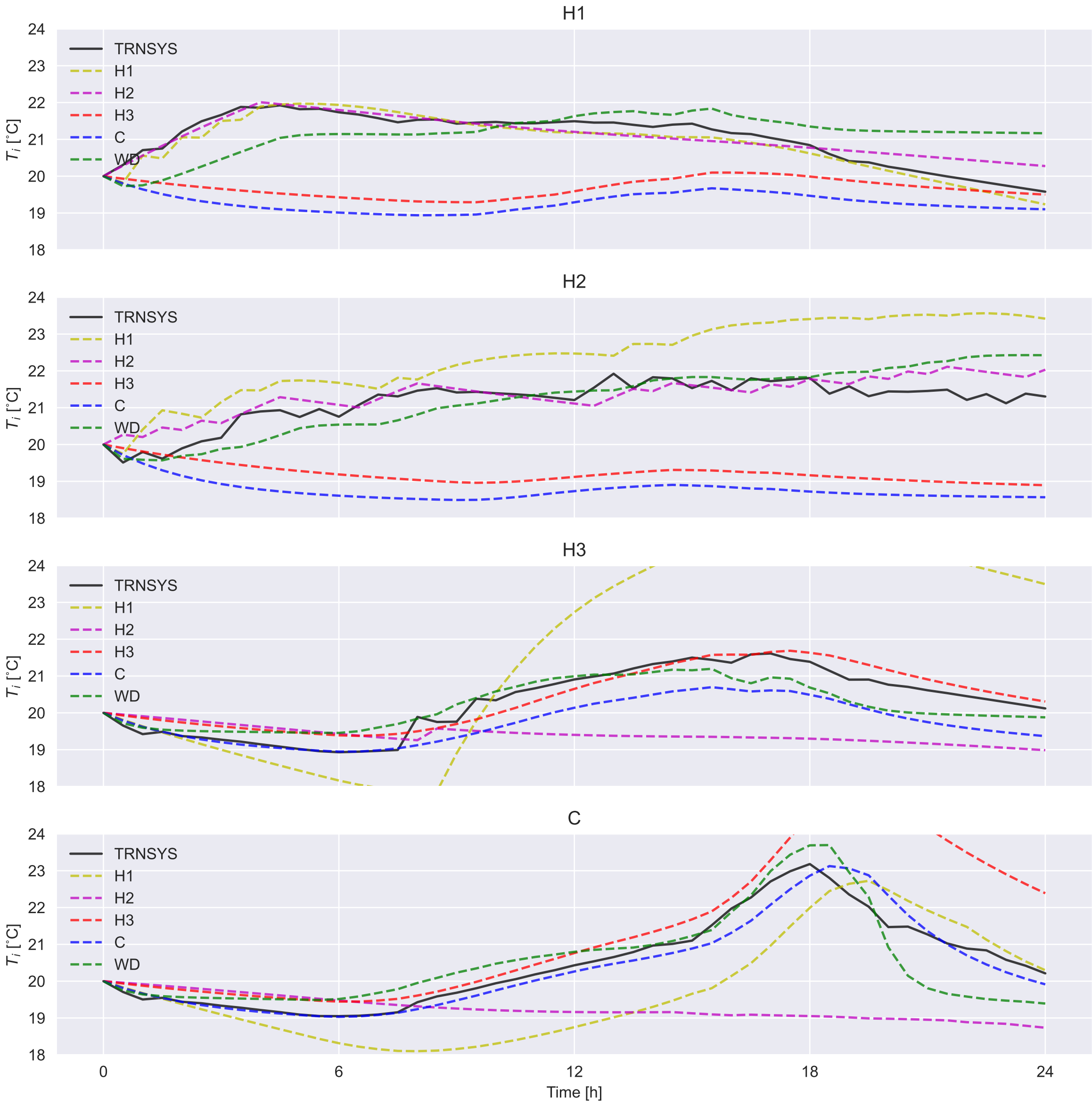

Figure 8: Qualitative results showing the performance of the models on different cluster scenarios. H1, H2, H3, C, WD refer to the model identified on the heating scenario 1, heating scenario 2, heating scenario 3, cooling scenario, and on the whole data set.

Perera, A. and P. Kamalaruban (2021, March). Applications of reinforcement learning in energy systems. *Renewable and Sustainable Energy Reviews 137*, 110618.

Refaeilzadeh, P., L. Tang, and H. Liu (2016). Cross-Validation. In L. Liu and M. T. Özsu (Eds), *Encyclopedia of Database Systems*, pp. 1–7. New York, NY: Springer New York.

Schilders, W. H. A., H. A. van der Vorst, J. Rommes, H.-G. Bock, F. de Hoog, A. Friedman, A. Gupta, H. Neunzert, W. R. Pulleyblank, T. Rusten, F. Santosa, A.-K. Tornberg, L. L. Bonilla, R. Mattheij, and O. Scherzer (edited by) (2008). *Model Order Reduction: Theory, Research Aspects and Applications*, Volume 13 of *Mathematics in Industry*. Berlin, Heidelberg: Springer Berlin Heidelberg.

Sinaga, K. P. and M.-S. Yang (2020). Unsupervised K-Means Clustering Algorithm. *IEEE Access 8*, 80716–80727. Conference Name: IEEE Access.

Waibel, C., R. Evins, and J. Carmeliet (2019). Clustering and ranking based methods for selecting tuned search heuristic parameters. In *2019 IEEE Congress on Evolutionary Computation (CEC)*, pp. 2931–2940. IEEE.