Applied Energy 228 (2018) 1439–1453

Contents lists available at ScienceDirect

Applied Energy

journal homepage: www.elsevier.com/locate/apenergy

Component-based machine learning for performance prediction in building design

Philipp Geyer*, Sundaravelpandian Singaravel

Architectural Engineering, Faculty of Engineering Science, KU Leuven, Belgium

HIGHLIGHTS

- A component-based method of machine learning for performance prediction in engineering.
- Components instead of one monolithic model extend reusability and generalization.
- Flexible design support by machine learning for early design phases.
- Internal parameters between components allow insights in the "black box" of machine learning.
- Good prediction accuracies (error < 3.9%) for test cases different from the training model.

ARTICLE INFO

Keywords:
Component-based machine learning
Systems engineering
Parametric systems modeling
Building performance prediction
Building simulation

ABSTRACT

Machine learning is increasingly being used to predict building performance. It replaces building performance simulation, and is used for data analytics. Major benefits include the simplification of prediction models and a dramatic reduction in computation times. However, the monolithic whole-building models suffer from a limited transfer of models and their data to other contexts. This imposes a vital limitation on the application of machine learning in building design. In this paper, we present a component-based approach that develops machine learning models not only for a parameterized whole building design, but for parameterized components of the design as well. Two decomposition levels, namely construction level components (wall, windows, floors, roof, etc.), and zone-level components, are examined. Results in test cases show that, depending on how far the cases deviate from the training case and its data, high prediction quality may be achieved with errors as low as 3.7% for cooling and 3.9% for heating.

1. Introduction

The challenge of a sustainable built environment requires the early integration of performance in design processes. This leads to a complexity never seen before in building design. To manage this complexity, designers, planners and engineers need to quickly obtain an overview of the overall performance of a building, including the systemic dependencies. Systemic dependencies often cross disciplinary boundaries, and thus lead to multidisciplinary interdependencies that play a vital role in overall performance. To manage this complexity, information is needed on performance, on these interdependencies, on the emerging design space, and on specific well-performing regions in a process of design space exploration (DSE) [1,2] in order to be available quickly enough and easily enough for the design process.

Machine learning (ML) provides a solution in this situation, offering the advantages of fast prediction and simplified parameter structures matching early design phases. This, allows designers and engineers to change designs quickly and to observe the consequences for performance in a DSE process. However, current ML approaches are developed too specifically for design situations, and require redevelopment for new cases.

The paper will therefore develop a component-based approach supporting such a systemic performance prediction. Based on a component-oriented structure, and on machine learning models, the thermal energy performance of buildings is predicted as an example of performance-based design. The aim is to prove the novel component-based approach of ML with components that serve prediction in multiple cases. Although the paper is limited to thermal performance, other disciplines of building performance can be treated in a similar manner, thus allowing a link to be created to observe systemic interdependencies and multidisciplinary building performance.

* Corresponding author.
E-mail address: p.geyer@kuleuven.be (P. Geyer).

https://doi.org/10.1016/j.apenergy.2018.07.011
Received 26 October 2017; Received in revised form 20 April 2018; Accepted 3 July 2018
Available online 13 July 2018

1.1. Background

Two approaches currently exist to predict performance in building design: firstly, physical modeling and simulation and, secondly, machine learning models. In terms of modeling categorization, the first approach is also known as white box modeling, whereas the second is known as black box modeling.

The methods of physical white box modeling of buildings are quite well examined, as is shown by the fundamental literature such as the books by Clarke [3] or the book by Hensen and Lamberts [4]. These approaches consider dynamic effects as they are required to provide quite precise results. However, they have a high information demand and involve relatively long computation times. Multi-domain simulation (co-simulation) can be carried out by coupling energy simulation with other simulation models, and has been shown in several examples [5–7]. This allows the consideration of discipline-crossing dependencies, thus permitting the complexity described above to be addressed. However, the effort involved in terms of modeling and computation time is significant, and exceeds what is possible in normal design situations.

Black box methods of statistical surrogate or meta modeling and machine learning offer effective energy performance prediction in this situation of complex design and engineering [8–10]. Such methods, also known as black box models, typically act in an analogous way to a physical simulation. Surrogate models based on the response surface method (RSM) were an early approach towards reducing the number of experiments by making good predictions [11,12].

The approach of black box modeling based on ML and surrogate modeling is used for building energy performance prediction, as is shown by reviews of the field [13–17]. Table 1 breaks the selected exemplary studies down into categories. Model Category 1 uses ML and other surrogate models to give static feedback, which is basically one number per feedback variable, such as yearly total heating energy demand in kWh/a. This is carried out for application at subsystem level, at building level and at urban level. For the latter two subcategories, parametric black box models of buildings are used to quickly predict energy consumption. This supports either one design, or the optimization of one building (Section 1.2). Furthermore, black box models have been applied to the design and optimization of subsystems (Section 1.1). Besides these applications with a static response, models with a dynamic response have been developed (Section 2). On the one hand, they are applied for the purposes of design and optimization, i.e. to identify peaks in operation quickly and to reduce them in designing (Section 2.1). The second purpose of developing dynamic-response models is application in control (Section 2.2). All these models are monolithic black box models, which means that they cannot interpret what is happening between input and output models in a physical way. For instance, monolithic building models predict the heating energy consumption by input parameters, such as physical properties of windows and wall. However, internal properties that lead to the energy prediction, e.g. the share of heat flows through the windows and walls, are not known due to the monolithic characteristic of the model. This missing internal model information is a significant disadvantage of black box models that is overcome by taking a component-based approach.

Table 1
Overview of relevant ML modelling approaches.

Model type	Application area	Studies
Monolithic model of static energy prediction (total yearly energy consumption)	Subsystem: Zone/HVAC/ Building component	[18–25]
	Building level: Design, Optimization	
	Building stock: Urban, Regional, National	[26–30]
Monolithic model of dynamic energy consumption patterns	Building level: Design, Optimization	[31–35]
	Building level: Control	[36–46]
Systems models	Energy systems/Systems engineering with ML models	[47–51]

When it comes to analyzing and designing energy systems, the application of ML takes place in a system engineering-based way that provides internal information and produces reusable system components (Section 3 in Table 1). This is founded on the discipline of systems engineering [52–55]. This is very valuable for designing engineering artifacts, helping to understand dependencies between its system modules, and for managing the related complexity, as it is performed by the design structure matrix method [56,57]. This method has been transferred to building construction and its adaptability [58,59], as well as to information flows during design [56,60,61]. Modeling approaches following this modular method have a high potential for understanding the complexity caused by sustainability, as is shown by the system modeling approach that has been developed for building design and urban contexts [62,63]. As is described in the next section, we follow this systems approach and use it as a basis to develop prediction models for the domain of building design.

1.2. Component-based approach

On this basis, we propose a component-based approach using machine learning for the prediction of performance and the management of complexity arising in the design and planning of energy-efficient and sustainable buildings. The component is a subordinate model of a building part or its technology, such as walls, windows, roof, floor slab as well as heaters, chillers, etc. It is defined by input and output parameters. In the interest of rapid prediction, the component-based approach uses surrogate models (black box models) to connect these parameters, and combines them with the paradigms of systems engineering to manage the complexity that arises. In this paper, we develop a component-based model using ML.

We expect four advantages to ensue from the component-based approach:

- In contrast to the monolithic use of surrogate models, it allows the potential of system engineering to be exploited for complexity management.
- By building models for general components, such as walls, windows, roofs, etc., and zones, we expect a much greater degree of generalization, i.e., transferability to other new cases not included in the training model structure, something which is currently a problem for black box methods in building performance prediction in design.
- The component-based approach links very well to building information modeling (BIM) [64–66] as an upcoming method of future design and planning.
- The component-based models make quantities available in the analysis between components. This design-supporting insight is made available by a detailed performance simulation, but not by monolithic black-box models. It enables designers and engineers to analyze and understand systemic interdependencies much better.

Basically, from the perspective of machine learning, component-based ML is an engineering-based application of deep learning [67,68] that includes transfer learning, something which is discussed in detail in an accompanying paper [69]. This also includes multilevel models within components, which in turn depend on the availability of data or on the need to achieve more accurate predictions.

The component-based method has been developed for energy performance prediction. However, the method is applicable to all types of simulation results to build surrogate models and to form a systems model predicting multidisciplinary performance for design space exploration. Within systems that model schemes, the components form

the linkage between design input variables and engineering-relevant performance output parameters. In the components, surrogate models describe the behavior in terms of performance.

Section 2 describes this method of systems modeling combined with the generation of ML component models. Systems modeling and ML need to be performed jointly because the decomposition depends not only on system engineering aspects, but also on parameters, data and the possible fit of components to training data. Section 2 provides a prototypical set of parameters that forms the basis for validation in Section 3. This section applies the developed ML components in test cases with the aim of validating their ability to also predict performance correctly in cases that go beyond the parametric structure of the training data, i.e. in cases that show different structures of components that cannot be covered by simply changing parameters within the simulation model for the training data, which would be equivalent to a monolithic surrogate model.

2. Method

This section describes how to decompose the design into a system of linked ML components using parametric systems modeling methods. For these components, parametric simulation serves to generate the training data for machine learning. Training the ML components based on a detailed parametric simulation allows the incorporation of a wide range of important effects that only simulation is able to predict correctly, such as the dynamic effects of thermal mass. The aim is to represent the results of the simulation in a manner that is easy to use in the form of the components. The simulation results therefore serve to train and test the ML model per component. In the next section, finally, components are put together to form a linked system that provides the desired performance for a design configuration.

2.1. Parametric systems modeling (PSM)

The component-based approach requires the decomposition and parametrization of the design. Relevant options to be changed in the design require consideration in the performance prediction model. There are two possibilities of varying a design: (1) Parameters of the design, such as length or width of the building, size of the windows, strength of insulation, etc., are changed; (2) The structure of the design is changed, i.e. walls, windows, doors, storeys, zones, etc., are added or removed. The second type of change plays a pivotal role in designing. It is very difficult to cover these changes using a parametric model, as the combinations of possible changes quickly become unmanageable. For this reason, we address this situation by taking the component-based approach. Decomposition, which is the first step, leads to a description of the design by parameterized components and their combination as a system structure. The performance of the design is aggregated by the behavior of the components based on ML models.

The component-based approach is founded on the parametric systems modeling method [62,63]. This method leads to parametric structures following a component-based systems engineering approach. The Systems Modeling Language (SysML) [70] facilitates a formal description of these structures using diagrams as shown for monolithic modeling and component-based modeling in Fig. 1. The parametric constraint blocks in this figure represent interfaces to fit the surrogate models, i.e. the ML model. They can represent building construction as well as building technology. A set of input parameters and the respective response or output determine the state of a component.

2.2. Decomposition

The component-based approach requires decomposition, which consists of breaking down the artifact into components. Decomposition requires the consideration of some criteria to deliver well-suited, suitably-generalizing models:

(1) In terms of **recurrence**, decomposition needs to identify basic reusable elements. This is key to generalization. We propose two different approaches in that respect: (a) the break-down follows building elements as they are normally used in designing, such as wall, window, door, roof, floor slab, etc., or (b) the design is broken down into zones that include their parameters for their enclosure properties.
(2) A **suitable parameter structure** is required to be able to adapt the components to the respective application situation. For these parameters, a suitable range and coverage by training data needs to be assured in order to capture the required parameter space.
(3) A **sufficient fit** of the surrogate models is required for the chosen decomposition and its component models. Whereas an a priori assessment can be carried out for the first two criteria before dealing with empirical material, this criterion requires an a posteriori evaluation in that at least the training data are known with their complexity, non-linearity, discontinuity, etc.
(4) **Integration into existing data structures** is highly relevant, as this allows easy access to required information and integration into design environments. Building information modeling (BIM), as it is currently emerging [64–66], is the basis to be considered. Section 2.3 discusses this aspect.

Fig. 2 shows the two structures that have been developed in a decomposition process. This process was iterative, including tests of how different machine learning models were suited to alternative structures of components and parameters, and how they matched the other criteria described above.

The parameters and the response of the components can be static or dynamic. For instance, energy consumption can be expressed as hourly energy consumption over a year, or total annual energy consumption. We have tested two configurations in the decomposition:

(a) **Static response components at construction element level:** The structure that resulted from this decomposition method (Fig. 2a) consists at the first layer of construction level components, such as walls, windows, floors, roof, etc. These components use only design parameters, such as the dimensions of the building, window-to-wall ratio or material properties, to predict the yearly total heat and

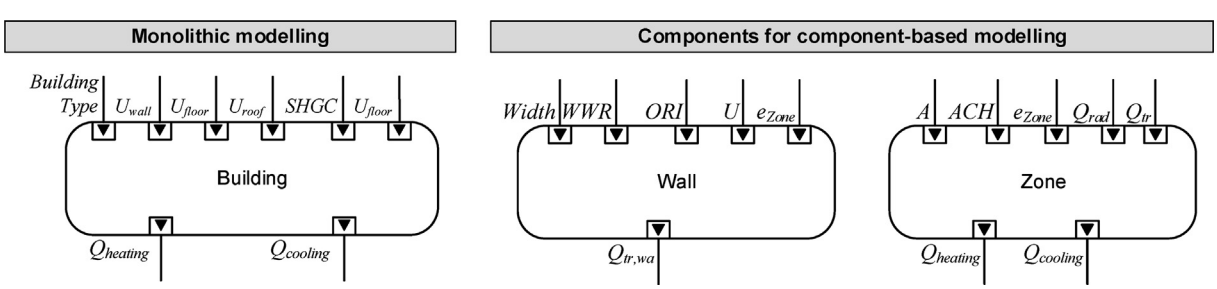


Fig. 1. Parametric diagrams of approaches of surrogate components according to systems modeling (see Fig. 2 for a legend).

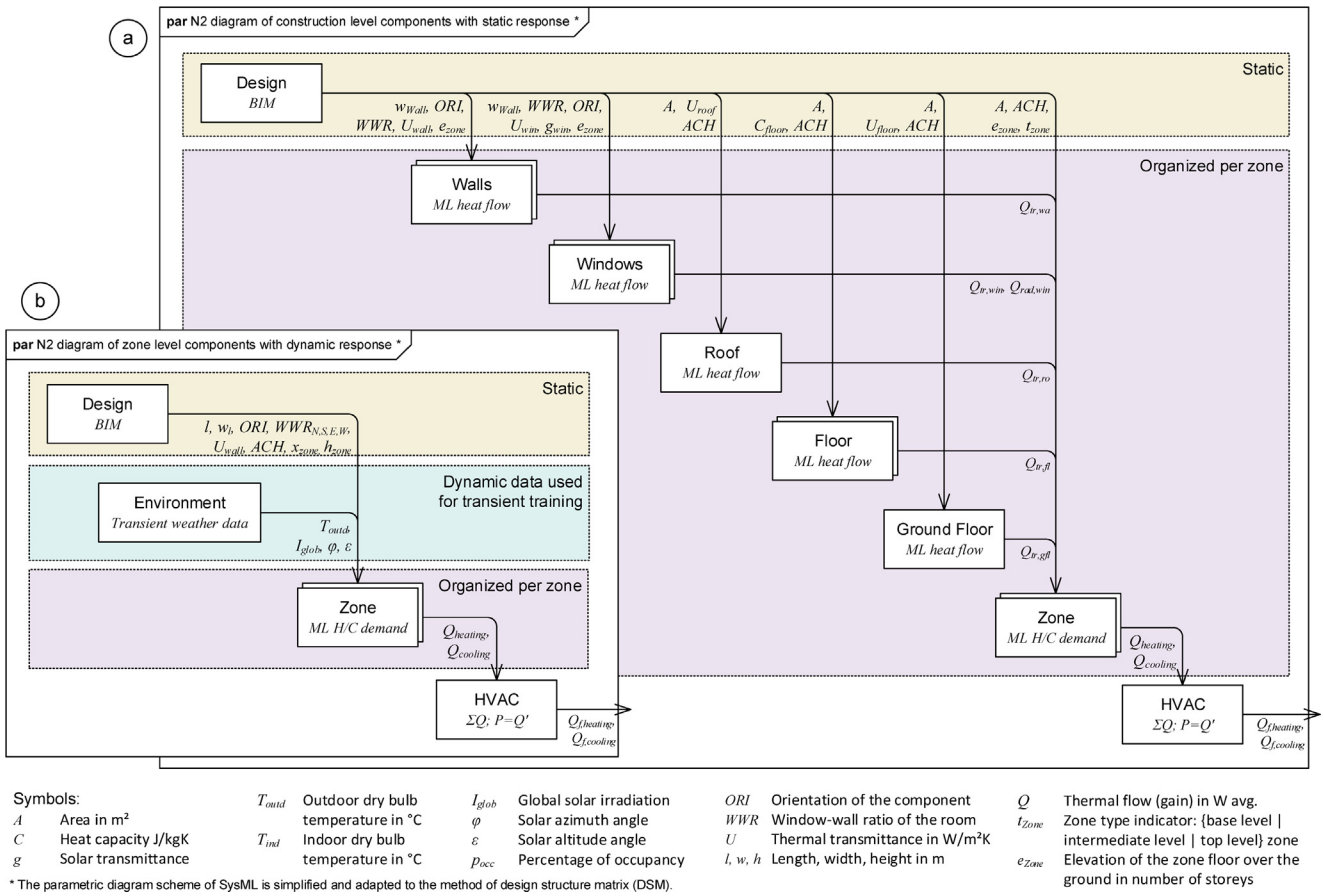


Fig. 2. Decomposition in an N^2 diagram (a) for static response with construction level components and (b) for dynamic response with zone level components.

radiation flows. The static response of these components forms the input for the zone level component at the second layer, which in turn consists of two ML models that transform transmission heat losses Q_{tr} and radiation gains Q_{rad} of previous components to heating and cooling loads $Q_{heating}$ and $Q_{cooling}$.

(b) **Dynamic response components at zone level:** The second structure has as a core element a zone ML model that delivers a dynamic monthly response (Fig. 2b). The inputs for this component are, firstly, the static design parameters and, secondly, dynamic weather data. Weather data is auto-correlated, i.e. weather data at a previous moment has an effect on current weather data. For example, January's weather pattern influences the month of February. The use of dynamic weather data in the prediction process adds complexity to the models, which makes ML training more difficult. The outputs of the ML model are monthly heating and cooling demands $Q_{heating}$ and $Q_{cooling}$. The response from each zone ML model is aggregated to form the monthly demand of the building.

Both responses, that is including the static ML model for the component's response, cover dynamic effects because the training data are based on a dynamic simulation. Dynamic effects, e.g. those of thermal mass, are therefore accurately predicted as far as the ML models are well fitted to the training data from parametric simulation.

The structure has been developed for a representative office building case. We therefore anticipate that it is valid for similar buildings in this category. We however expected to come up against limitations of this structure, such as the multi-storey structure, which would not suit a hall building. In contrast, residential buildings could probably be described with the same structure, but with another set of training data.

2.3. ML modeling strategy

We use artificial neural networks (ANN) to represent the components' behavior. This ML method has a high degree of flexibility when it comes to representing data regression. Nonetheless, there might also be other methods for properly representing the component's behavior and response.

2.3.1. The static ML model

The development of ML components representing construction elements uses neural networks with one hidden layer per component with between 10 and 20 neurons, based on a sigmoid activation function. Fig. 3 shows the architecture of the neural networks for the components as well as the most important connections between the components for the static prediction. The architecture implements the developed decomposition shown in Fig. 2. The input parameters, the number of hidden neurons, and training algorithms for the components, have been selected in a process of feature engineering, based on engineering experience and on observations made in training, i.e. reduction of component cross validation errors. There are complex phenomena, such as radiation in the case of windows, requiring more hidden units and advanced training strategies, whereas other components such as the roof or the floor slab rely on simpler thermal dependencies, and thus can be represented by simpler structures using basic training algorithms.

2.3.2. Dynamic model

Besides the static building component models, a dynamic model has been developed for a parameterized thermal zone. This model delivers monthly heating and cooling energy demand based on a thermal zone. Fig. 4 shows the architecture of this ANN model. The model input

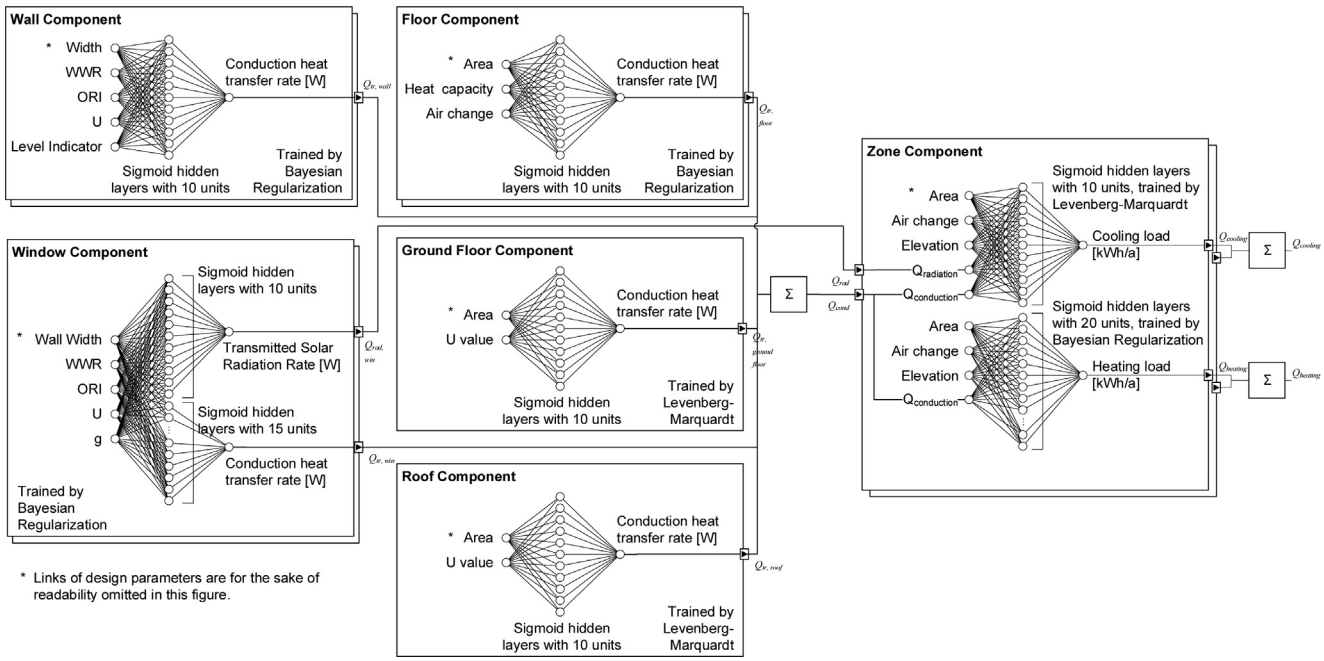


Fig. 3. Neural network architecture for components for static prediction.

parameters are monthly average weather data and design parameters. Time dependence of the energy demand predictions calls for inputs at a specific month based on historic information. Use of long short-term memory (LSTM) in the ML model enables the storage of historic predictions and their deployment to make current predictions. The use of the LSTM layer in the model allows the ML model to capture dynamic interactions in the weather data with the energy demand of a thermal zone. The LSTM layer training time is higher than simply hidden layers. The LSTM layer for heating and cooling predictions are hence shared in this paper. This allows us to train a model with two output layers and one hidden LSTM layer.

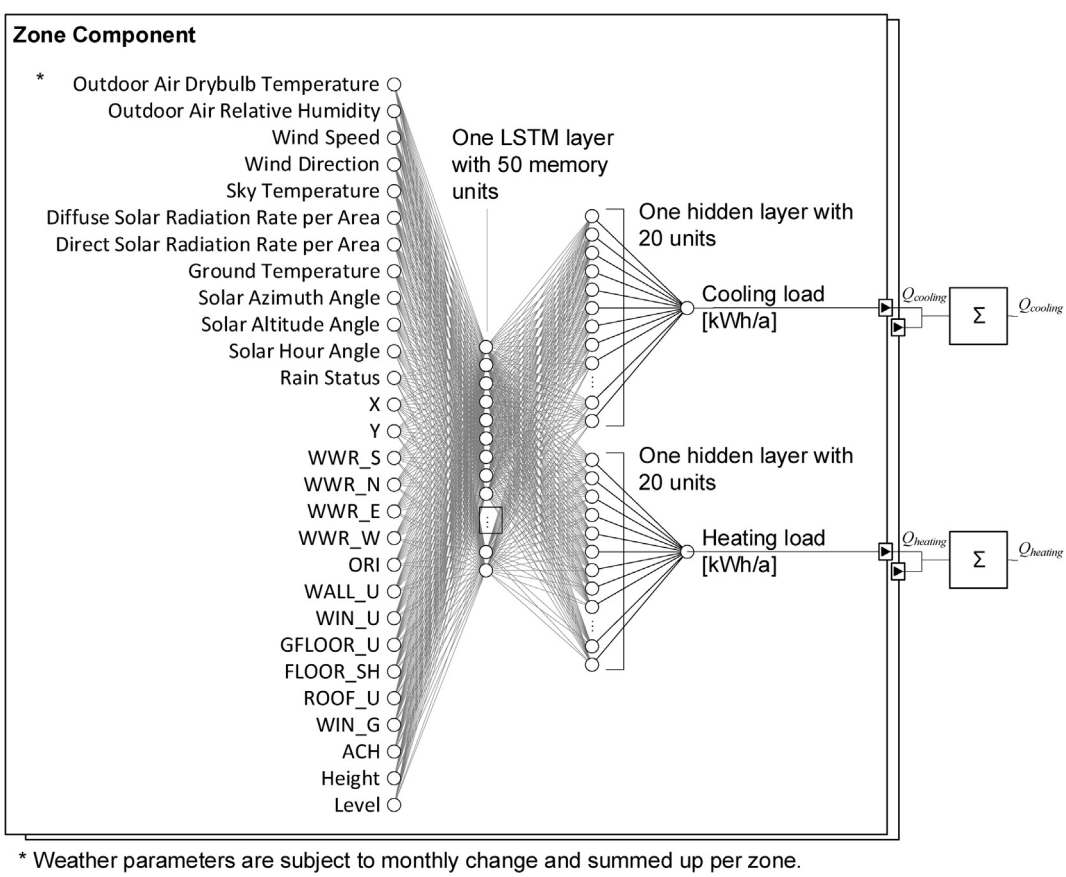
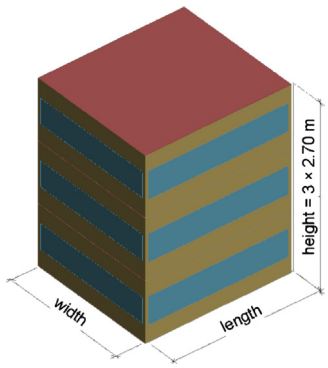


Fig. 4. Dynamic model structure with LSTM layer to incorporate dynamic effects.



		Unit	min	max
Length		m	4	80
Width		m	4	80
Window-to-wall ratio (WWR)	S		0	0.95
	N		0	0.95
	E		0	0.95
	W		0	0.95
ORI		°	-180	180
U value walls		W/m2K	0.411	0.776
U values windows		W/m2K	0.5	2
U value ground floor		W/m2K	0.411	0.864
Heat capacity floors		J/kgK	900	1200
U value roof		W/m2K	0.191	0.434
g value windows			0.5	0.95
Air change rate		h^{-1}	0.2	1
Level height ground floor zone		Storeys	0	10

Fig. 5. Left: Parametric simulation model implemented in the Energy Plus software for the generation of training data for the components; Right: Parameters and ranges of the training data generated by this model.

2.4. Training data from a parametric simulation

A key prerequisite for ML is constituted by representative training data. Parametric simulations implemented in the Energy Plus software served to generate the training data. This training model consists of a three-storey building (ground floor + two storeys) in that its design parameters are varied to cover a representative design space. Assumptions made in this model are the use of ideal HVAC loads, adiabatic internal floors and fixed internal gains (10 m^2 per person for occupancy, 10 W/m^2 for lighting and 15 W/m^2 for equipment).

The parametric model sampled 800 design combinations by using a Latin Hypercube sampling scheme. Fig. 5 shows the list of sampled design parameters with the training data ranges (right) and a visualization of the simulation model (left). The sampling process consisted of two parts. The first part included 400 design combinations that have windows in one orientation only. This was required to cover the effect of windows in each orientation separately in order to ensure that the data was representative in this respect. The remaining 400 design combinations are based on the variation of all the design parameters. Table 2 shows exemplary data from the most recent batch of design combinations used for the training of the wall component.

Representative coverage of the design space is not only required in terms of the range of each individual parameter, but also for parameter combinations that are connected to each other. It therefore turns out to be useful to focus the parametric combinations on regions that really occur in the design process, as other combinations would produce misleading training data that could hinder the training process of the ML model. One important connection in this case was the combination of building length *l* and building width *w*. Not all possible combinations of these parameters will occur in real designs, such as very slender floor plans. Such combinations have been excluded from training, and selective sampling has been developed, as is shown in Fig. 6. The training data distribution in this sampling is tailored to the expected occurrence of geometric building dimensions in the prediction cases. This step permits an improved training process to be achieved, as confirmed by observations in the component training.

2.5. Validation strategy

The ML model development uses results from the computer experiments run in the Energy Plus physical building simulation software. The ML models developed rely on the validation of the energy prediction capabilities of this software against physical experiments, as is shown in its engineering reference [71] and in further testing and validation efforts [72]. Besides these efforts, extensive efforts to validate the software are undertaken by the International Energy Agency's Building Energy Simulation Test and Diagnostic Method (IEA-BESTEST) [73–75]. A correct prediction of the physical effects by the simulation software is therefore assumed.

The validation strategy in this research thus focuses on a comparison of the results with a physical simulation. There are two validation steps:

(1) In the development and training ML component, cross validation serves for validation at component level for validation. Training data are split into a set used in the training process (85%) and a set used for independent cross validation of components after testing (15%). This is described in greater detail in the training process in Section 2.6.
(2) After component training, test cases are analyzed with the ML components and with the methods of the Energy Plus physical simulation software. The results are compared in order to examine whether the ML prediction matches the simulation. This is described in Section 3.

2.6. Training of machine learning models for components

The training data serve as a basis to fit individual machine learning component models. This section describes the fitting process with the specific conditions that emerged. A model is matched for each of the components shown in Fig. 2. According to the two modeling approaches mentioned in Section 2.2, two different ML modeling and fitting processes emerge: (1) the detailed construction level component approach that predicts yearly total energy demand (Section 2.6.1), and (2) the dynamic response modeling approach that works with the thermal zone as a component for the monthly prediction (Section 2.6.2).

2.6.1. Static response ML model development

Before the training of the models was possible, some preparation of the input data was required. First, for walls and windows, the parametric simulation results include components oriented in four different directions, i.e. walls in the North, South, East and West. To exploit the full range of training data, all four wall responses are considered in the training data with a corrected orientation parameter. As a result, the wall component has a higher density of points (Fig. 7, top left). The same process was used for windows, with the difference however that only the first 400 samples were usable, as no orientation-specific response was available in the simulation results. For the other components, such as roof, floor slab and ground floor, training data are used directly from the data set. Furthermore, by adding indicators of

Table 2
Exemplary training data for the wall component.

X: Design Variables for Wall Component Training								Y: Wall Conduction Heat Transfer Rate [W](Monthly)															
$X	$Y	$WWR_S	$WWR_N	$WWR_E	$WWR_W	$ORI	$WALL_U	Level 1				Level 2				Level 3			
								South	West	North	East	South	West	North	East	South	West	North	East
m	m					°	W/m²K	W	W	W	W	W	W	W	W	W	W	W	W
77.48	32.2	0.115	0.857	0.749	0.133	173.2	0.325	−16499	−6520	−2440	−1791	−19119	−7590	−2879	−2111	−17425	−6907	−2612	−1916
28.18	25.72	0.393	0.933	0.14	0.279	172.8	0.373	−4948	−5213	−498	−5884	−5954	−6307	−612	−7223	−5534	−5855	−567	−6696
21.94	37.76	0.485	0.824	0.509	0.212	−140	0.376	−3405	−8371	−1047	−5362	−3825	−9519	−1199	−6073	−3565	−8851	−1114	−5652
21.35	27.07	0.337	0.432	0.582	0.243	−154	0.287	−3030	−4139	−2358	−2291	−3272	−4504	−2582	−2495	−3115	−4283	−2455	−2373
45.57	56.82	0.551	0.311	0.847	0.341	−115	0.383	−6115	−10452	−8645	−2593	−7181	−12460	−10342	−3048	−6714	−11630	−9652	−2850
22.49	57.5	0.258	0.895	0.835	0.806	−159	0.337	−4595	−2895	−589	−2445	−5195	−3307	−678	−2799	−4994	−3177	−652	−2689
15.94	19.96	0.766	0.141	0.94	0.309	19.54	0.294	−692	−2653	−2819	−234	−765	−2917	−3067	−257	−736	−2805	−2951	−247
22.11	51.58	0.723	0.016	0.308	0.529	−144	0.296	−1155	−4260	−3675	−6400	−1416	−5321	−4635	−7954	−1268	−4743	−4126	−7104
40.07	39.59	0.448	0.081	0.85	0.494	100.6	0.31	−4757	−4633	−8121	−1247	−5669	−5435	−9615	−1494	−5252	−5046	−8912	−1383
45.81	78.83	0.255	0.139	0.442	0.164	88.08	0.352	−8154	−17135	−9884	−10392	−9716	−20052	−11643	−12411	−9011	−18645	−10810	−11504
23.88	14.32	0.882	0.325	0.454	0.469	−163	0.29	−602	−1547	−3143	−1570	−671	−1735	−3556	−1766	−647	−1675	−3433	−1705
21	55.92	0.158	0.537	0.837	0.094	119	0.356	−4481	−13606	−2442	−2205	−5178	−15551	−2824	−2575	−4739	−14263	−2583	−2350
46.81	48.77	0.36	0.813	0.72	0.925	−22.2	0.278	−4907	−593	−1547	−2400	−5865	−711	−1822	−2826	−5507	−667	−1713	−2656
55.62	52.99	0.705	0.599	0.464	0.817	101.2	0.342	−4100	−2607	−5697	−6919	−4769	−2989	−6590	−8086	−4460	−2801	−6164	−7557
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

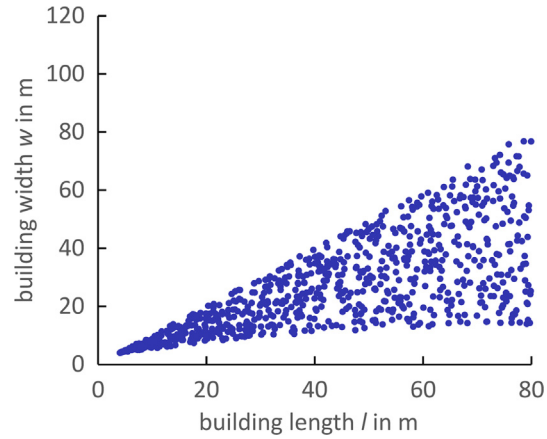


Fig. 6. The distribution of dimensions of the training building has been adapted to configurations occurring in the generation of the training data for the geometric ratio of the floor plan.

elevation and type of a zone e_{Zone} and t_{Zone} that provides information about the zone to which the window or wall is attached, the level of accuracy indicated by the coefficient of determination R^2 could be improved from 0.981 to 0.993 in case of the static response model of the wall component.

The static model development used simple one-layer artificial neural networks (ANN) per component. Two algorithms are used for training, as they are implemented in MATLAB: (1) the Levenberg-Marquardt algorithm and (2) Bayesian regularization. The first algorithm was used for the less complex datasets, whereas the latter was used for complex datasets, which are the walls, the windows, and the zone heating response due to interactions between large numbers of parameters. Most of the models achieved high levels of accuracy in terms of R^2 close to 0.99, or higher in a test with 200 independent random samples of the design space, as shown in Fig. 7. The model with the lowest R^2 is the floor model. However, this model has very low response values in the simulation (less than 300 kWh/a), which causes irregularities to dominate design parameters and makes the component subordinate in performance prediction. Errors sum up in an unfavorable way for some configurations in the zone heating model, leading to a few outliers and an R^2 of 0.971 for the heat demand of the zone. The resulting maximum test error for an independent test set with 200 random samples at component level is $1.67{\cdot}10^4$ kWh/a, which equals 9.3% of the maximum heat demand of the zone ($1.80{\cdot}10^5$ kWh/a), this being the worst response among all the models. For cooling, the maximum test error is 2.13·105 kWh/a, which is 6.0% of the maximum cooling demand of the zone ($3.56{\cdot}10^6$ kWh/a).

2.6.2. Dynamic response model

The thermal zone component is developed in Python using Keras library with TensorFlow backend. Adam training algorithm is used to develop the model. This is a first-order gradient-based optimization algorithm. The data generated is split into training, cross-validation and test data in a ratio of 70/15/15. Training data is used to develop the ML model, while cross-validation data is used to tune the model structure. Test data is used to evaluate the generalization. The resulting model has a test R^2 of about 0.99 for both heating and cooling energy predictions (Fig. 8). The results indicate that the developed models do generalize well on unseen data. The resulting maximum test error for monthly heating prediction is $2.34{\cdot}10^3$ Wh/month, which equals 6.17% of the maximum heat demand of the zone ($3.79{\cdot}10^4$ Wh/month), this being the worst response among the models. Whilst the test error for monthly cooling prediction is $4.5{\cdot}10^3$ Wh/month, this equals 3.12% of the maximum cooling demand of the zone ($1.46{\cdot}10^5$ kWh/month).

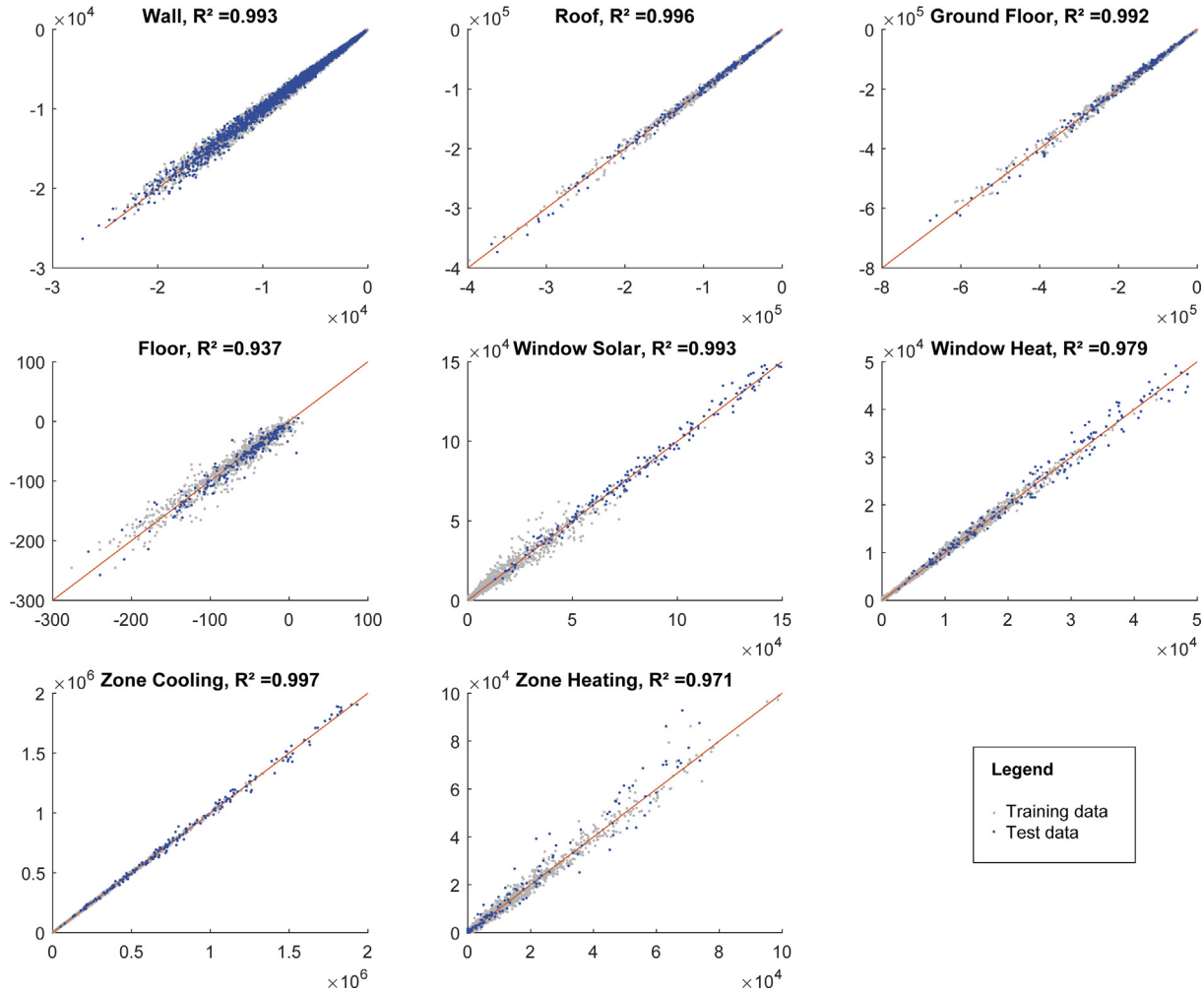


Fig. 7. Fitting of the static response components models to training data from a parametric simulation.

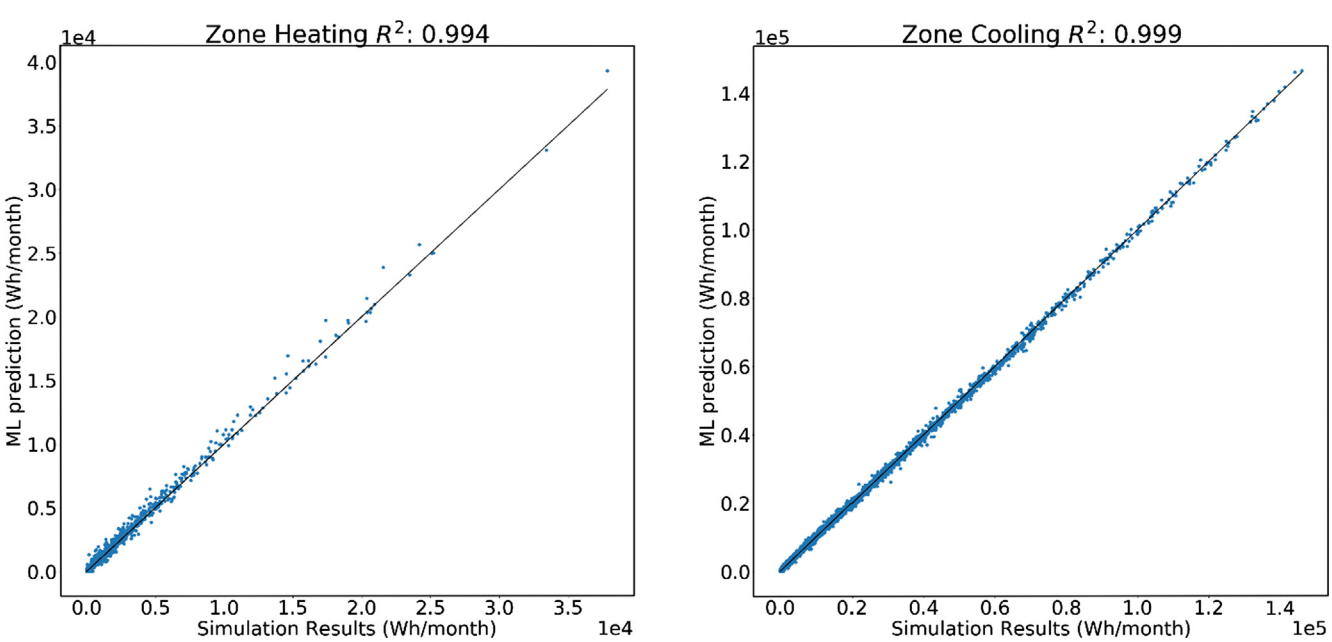


Fig. 8. Fitting of the dynamic response components models to training data from parametric simulation.

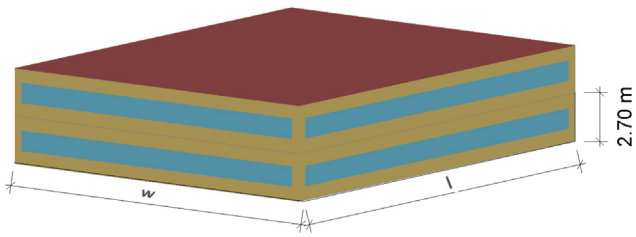


Fig. 9. Parameterized simulation model for Test Case 1.

3. Whole-building test cases

This section describes the validation of the component-based ML model at a building level. For this purpose, we developed independent simulation models for selected cases and compared the ML prediction with the simulation results. The aim is to test the components on cases that are representative of the building category for which they are developed. To achieve this, we used the design topology proposed by Dogan et al. [76] with partly simplified geometry, taking into consideration the box building-based training data.

3.1. Case 1: Two-storey rectangular building

This first case describes a rectangular office building configuration. The size and the envelope's physical properties have been parameterized to allow a more extensive validation. Fig. 9 shows the basic layout. The parametrization changes the length, width, window-to-wall ratio, U and g values as well as infiltration in the ranges indicated in Section 2.1. As a result of the parametrization, the case represents a rectangular building with two-storeys, and covers archetypes B1, D1 and E1 as well as A3 to E3 in the topology scheme of Dogan et al. [76] (Fig. 2), with one zone per storey.

Generating a test dataset of 200 random samples for this case and comparing simulation results with the ML prediction showed that the prediction was highly reliable. The comparison of the static response model in Fig. 10 shows very good matching of the ML prediction with the detailed building simulation, with a coefficient of determination R^2 of 0.948 for heating, and of 0.991 for cooling (Fig. 10). The maximum deviation for heating is $9.01 \cdot 10^3$ Wh/a, which is 5.1% of the maximum response ($1.74 \cdot 10^5$ Wh/a), and the maximum deviation for cooling is $1.06 \cdot 10^5$ Wh/a, which is 4.7% of the maximum response ($2.27 \cdot 10^6$ Wh/a).

Fig. 11 shows the ML predictions for zone monthly heating and cooling demand. Heating predictions have an R^2 of 0.692, and cooling predictions have an R^2 of 0.986. The maximum error for heating is $1.54 \cdot 10^4$ Wh/month, with a maximum heating demand of $2.85 \cdot 10^4$ Wh/month. This results in a maximum heating prediction error of 54%. The maximum error for cooling is $2.58 \cdot 10^4$ Wh/month, with a maximum cooling energy demand of $1.70 \cdot 10^5$ Wh/month, resulting in a maximum error of 15%. The cooling predictions have a good fit. However, the heating prediction does not have a good fit, the reasons being that the height of the top floor from the ground is higher in the training data than in the test case, and that the training data is predominately dominated by cooling, i.e. the case is slightly outside the training data. Incorporating data which covers more design space and heating demand will improve its performance.

3.2. Case 2: Eight-storey box building

The second test case deals with a rectangular high-rise building. It is composed of one ground-floor storey, six intermediate storeys and one top storey (Fig. 12). The elevation e_{Zone} and the type of the zone t_{Zone} serve to adapt the zone models to the respective situation. This test case is a simplified version of the archetype A5 in Dogan et al. [76] (Fig. 2), with box-like zones neglecting the free-form configuration. The same parametrization was applied as in the first test case. The 200 independent random samples delivered the same good match of static response, with an R^2 of 0.974 for heating and of 0.999 for cooling (Fig. 13). The maximum deviation for heating is $1.61 \cdot 10^4$ Wh/a, which is 3.7% of the maximum ($4.35 \cdot 10^5$ Wh/a), and the maximum deviation for cooling is $4.30 \cdot 10^5$ Wh/a, which is 3.9% of the maximum ($1.10 \cdot 10^7$ Wh/a).

Fig. 14 shows the monthly zone heating and cooling predictions of the ML model in Test Case 2. The R^2 for heating is 0.848, and the R^2 for cooling is 0.983. The maximum heating error for Test Case 2 is $1.54 \cdot 10^4$ Wh/month and a maximum heating demand of $4.02 \cdot 10^4$ Wh/month, resulting in a maximum heating error percentage of 38%. The maximum cooling energy error is $2.77 \cdot 10^4$ Wh/month, and the maximum cooling demand is $1.84 \cdot 10^5$ Wh/month, resulting in a maximum cooling error of 15%. A similar trend is observed as was seen in Test Case 1. However, the heating prediction errors are lower for Test Case 2 compared to Test Case 1. This is because this test case partly falls under the training distribution.

3.3. Case 3: Complex design

The third test case represents an architecturally slightly more advanced design in order to demonstrate that the method and the trained models are also fit for purpose in this case. Whereas Cases 1 and 2 included changes at storey level, this case includes different component configurations within one storey. By this feature, it demonstrates the capabilities of the component-based approach in this situation. The case has three storeys that do not have the same zone size. The lower two

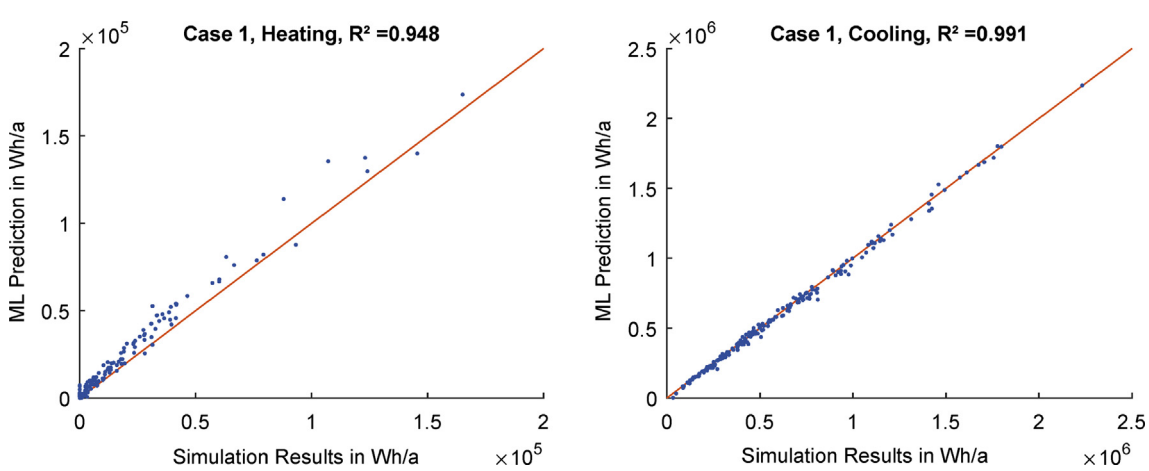


Fig. 10. Static response ML model: Validation results at building level for Test Case 1.

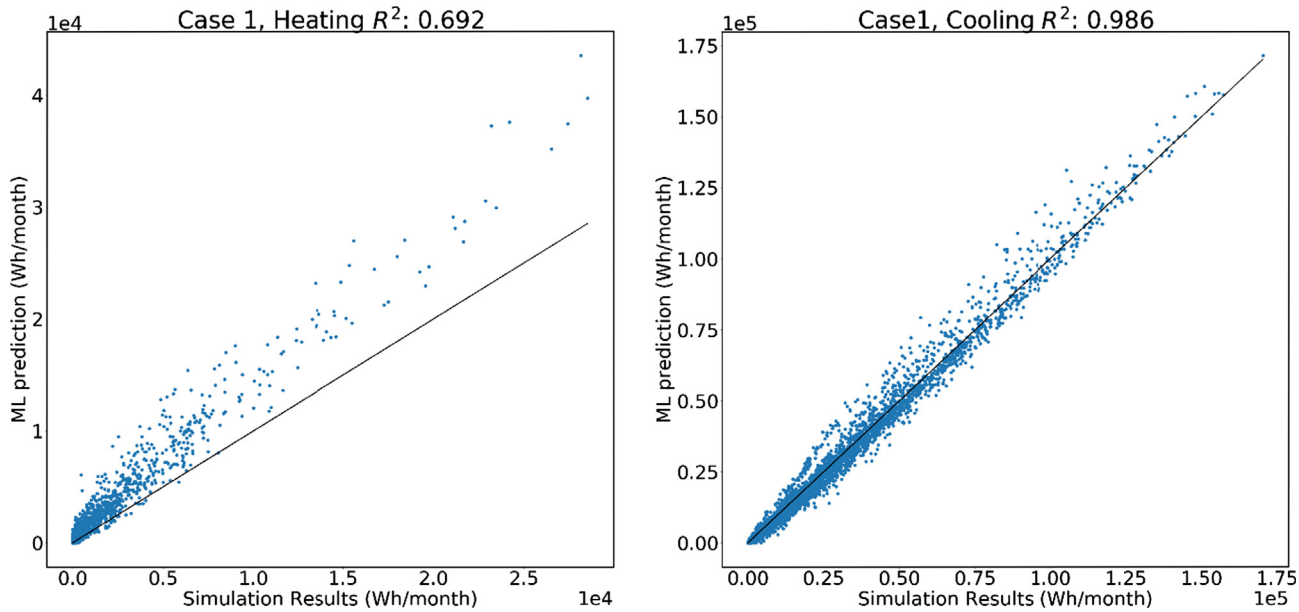


Fig. 11. Dynamic response ML model: Validation results at zone level for Test Case 1.

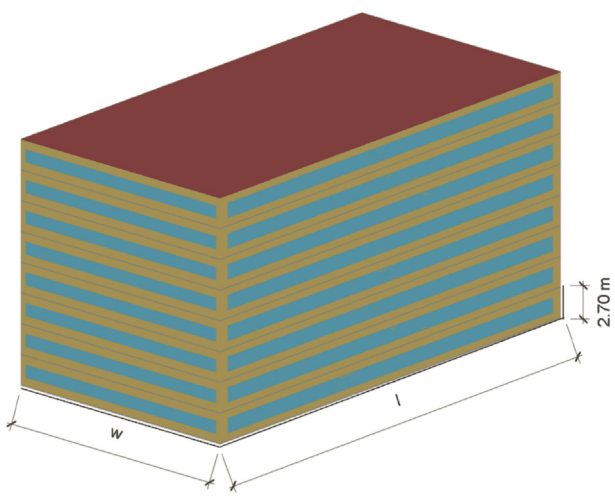


Fig. 12. Parameterized model for Test Case 2.

storeys therefore have a partial area covered by a roof to protect against outside conditions. The top storey is reduced by half the length of the building, and the base storey is extended by half of the building width, as is shown in Fig. 15, on the left.

The variants with the 200 random samples for the test cases show a reduced but sufficient accuracy of prediction, with an R^2 of 0.848 for heating and 0.982 for cooling (Fig. 16). The maximum deviation for the static response for heating is $7.2{\cdot}10^4$ Wh/a, or 29% of maximum heat demand ($2.5{\cdot}10^5$ Wh/a); for cooling it is $2.2{\cdot}10^5$ Wh/a, or 6.9% of the maximum ($3.36{\cdot}10^6$ Wh/a). Given the situation that a component configuration of a ceiling consisting half of roof and half of a floor slab is not part of the training data, the results validate the component-based approach.

Fig. 17 shows predictions of zone monthly heating and cooling demand. The heating energy prediction has an R^2 of 0.874, and the cooling energy prediction has an R^2 of 0.89. For Case 3, the maximum heating error is $1.39{\cdot}10^4$ Wh/month, with a maximum heating demand of $5.17{\cdot}10^4$ Wh/month. This results in a maximum heating error percentage of 27%. The maximum cooling error is $8.65{\cdot}10^4$ Wh/month, with a maximum cooling energy demand of $1.98{\cdot}10^5$ Wh/month, resulting in a maximum cooling error of 44%. In this test case, cooling energy R^2 is lower than in the other test cases, the reason being that the

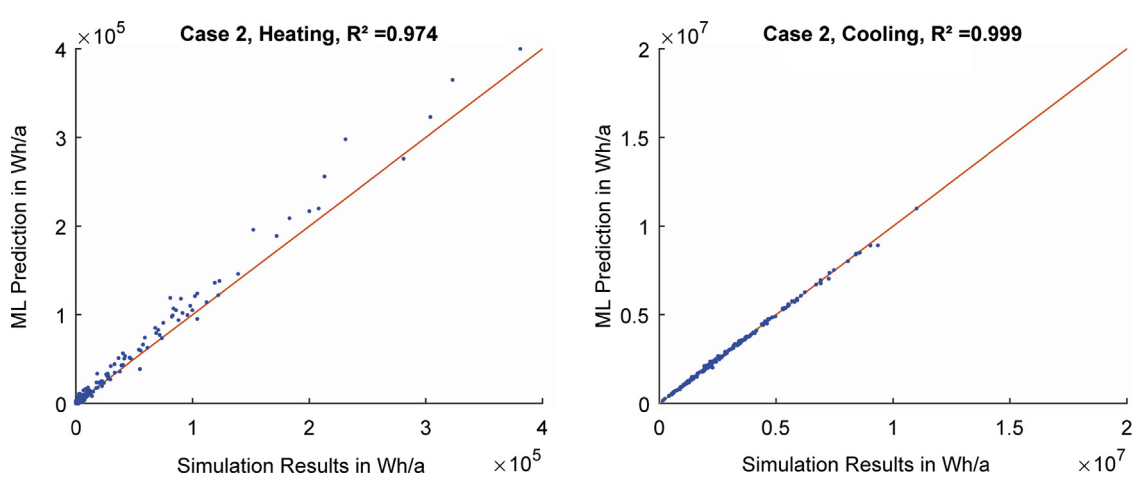


Fig. 13. Static response ML model: Validation results at building level for Test Case 2.

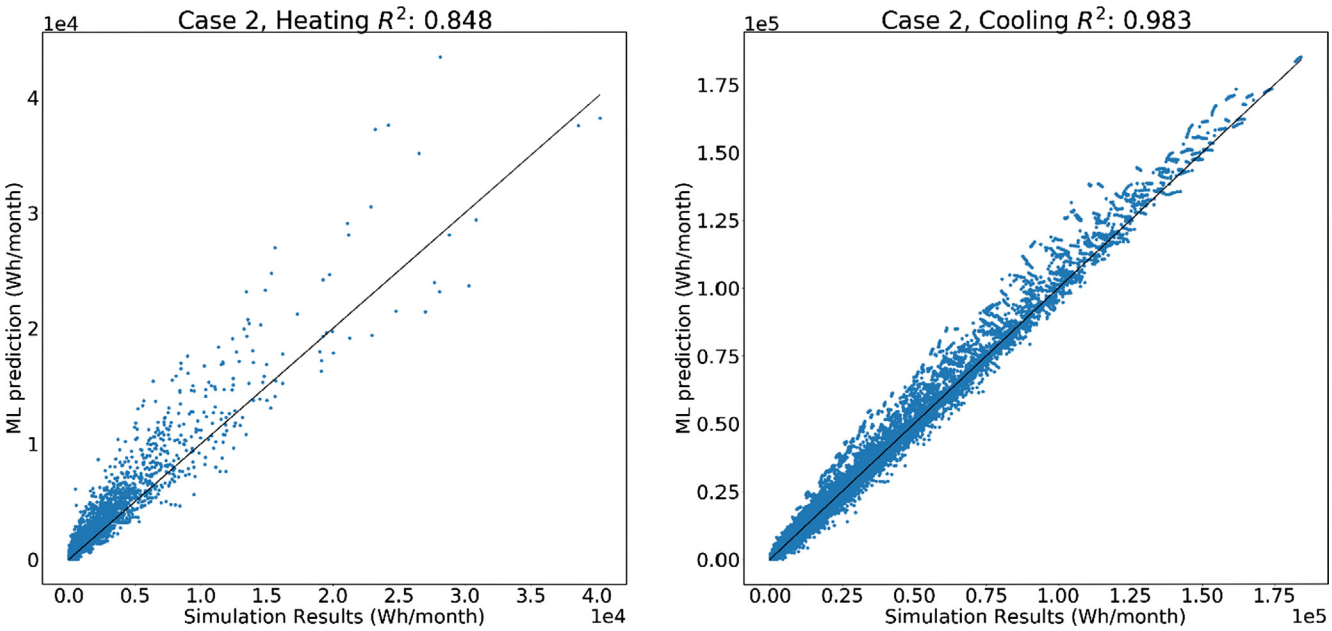


Fig. 14. Dynamic response ML model: Validation results at zone level for Test Case 2.

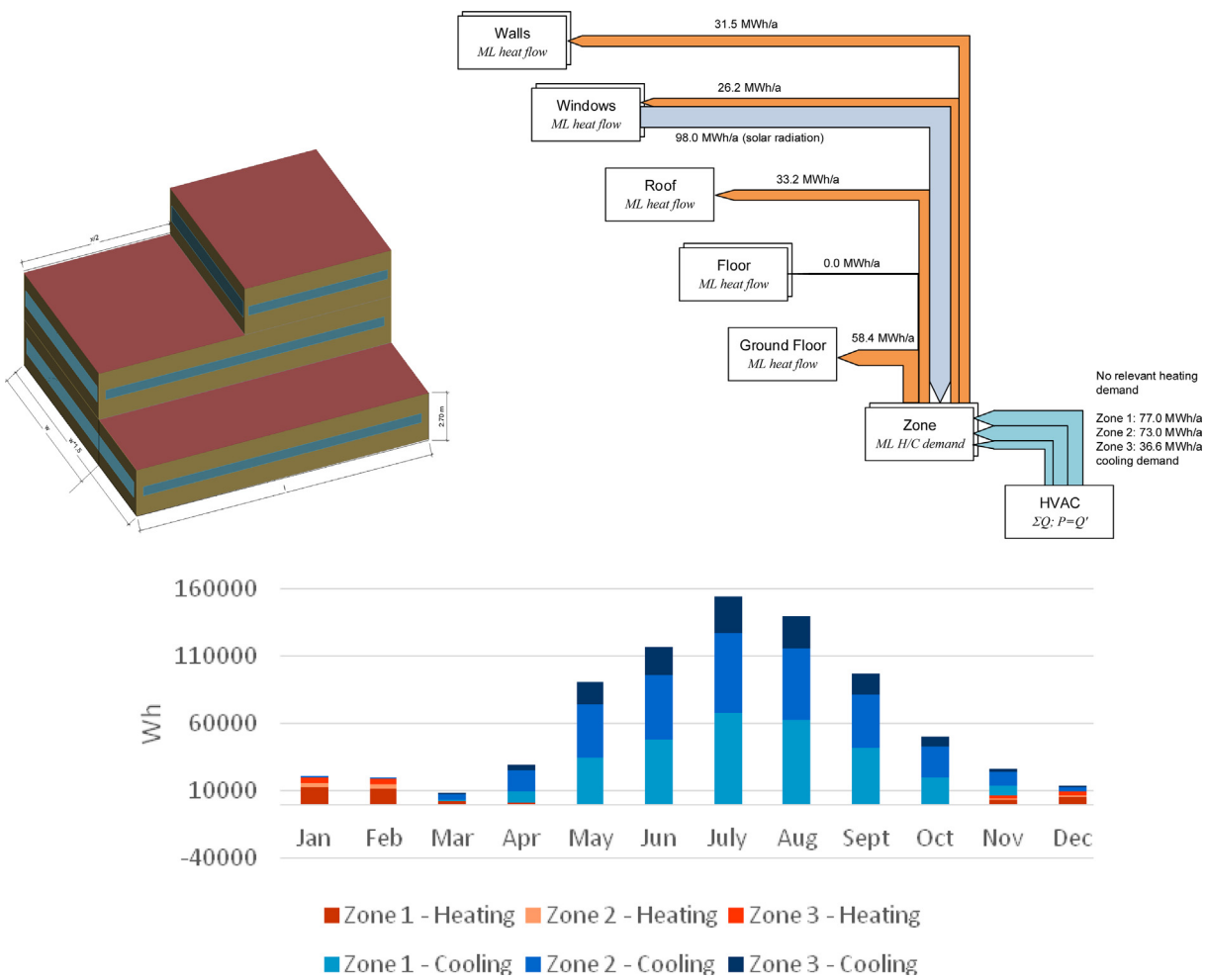


Fig. 15. Left: One configuration of Test Case 3. Right: Flow diagram illustrating internal quantities. Bottom: Illustration of monthly zone monthly energy demand.

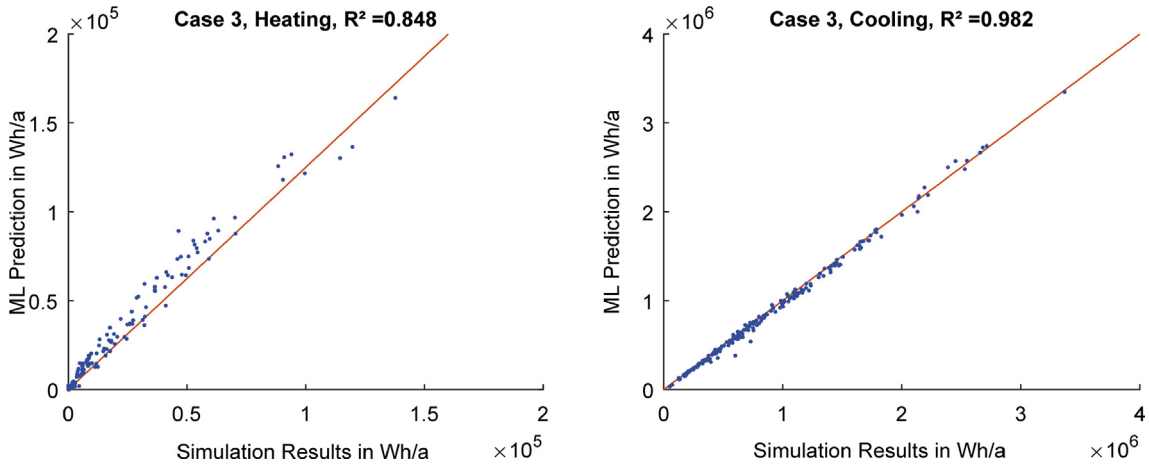


Fig. 16. Static response ML model: Validation results at building level for Test Case 3.

training data on zone level does not cover exposed ceilings for ground floor and first floor.

An important benefit of the component-based approach in contrast to the whole-building ML approach is that information on inter-component quantities is available for the design and analysis process. This is illustrated by the heat and radiation flows for one configuration by the diagram contained in Fig. 15, on the right. Fig. 15, at the bottom, illustrates the zone-wise monthly heating and cooling demand prediction.

Furthermore, Table 3 shows the Pearson correlation for the data in the 200 test samples in the static response as an indicator of interdependencies of intercomponent flows on design variables. This table allows for interpreting the effect that variables have on components. For instance, there is a high effect of the building dimensions X and Y on all flows, which is to be expected but quantified by the table. This effect is either positive, which means that flow increases with size, or it is negative. Furthermore, for instance, the effect of the window-to-wall ratio on window, wall and zone response is shown. As WWR increases, (negative) wall heat losses are reduced (shown by the positive coefficient), and window heat losses are increased. Moreover, radiation gains also increase.

Fig. 18 shows the errors observed for all three cases for the static and dynamic models. This overview shows that the static prediction model has a lower error in general than the dynamic prediction model. The dynamic prediction is coupled to a higher level of complexity, covering more discontinuities, such as month with and without heating/cooling demand. Furthermore, the prediction of the heating demand is coupled to a higher error since the degree of discontinuities of heating is higher in the specific configuration than that of cooling. We expect this to differ in other configurations of design parameters and parameters of internal loads. Finally, a slight but limited increase in inaccuracy is observable from Case 1 to Case 3, which means with increasing complexity of the design cases and with a deviation from the initial training case. The slight increase is a positive indication of the extensibility of the component-based modeling approach.

4. Discussion

With the ML models that have been developed, as well as the test cases, we have shown the feasibility and potential of a component-based approach of machine learning and its limitations. The application of ML for predicting energy performance has several advantages, but

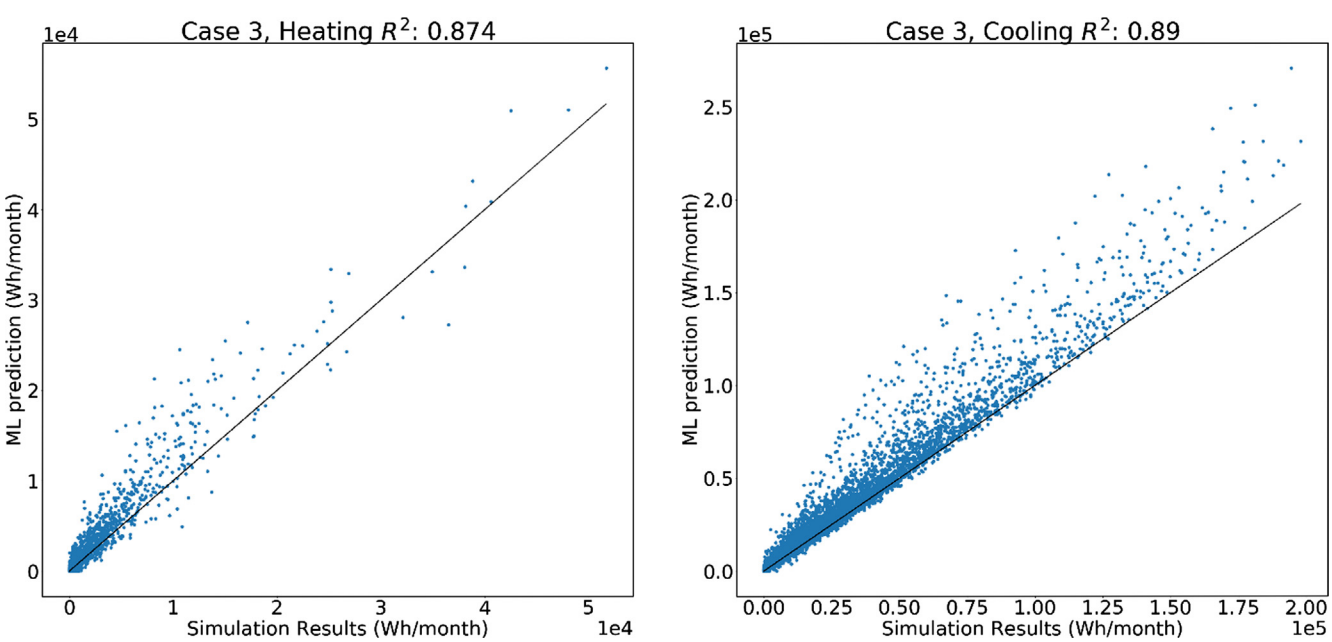


Fig. 17. Dynamic response ML model: Validation results at zone level for Test Case 3.

Table 3
Interdependencies of inter-component heat flows on design variables shown by the Pearson correlation (orange: positive corelation; blue: negative corelation). (For interpretation of the references to colours in this table legend, the reader is referred to the web version of this paper.)

		Wall heat flow	Window heat flow	Window radiation gain	Ground floor heat flow	Internal floor heat flow	Roof heat flow	Zone 1 heating demand	Zone 1 cooling demand	Zone 2 heating demand	Zone 2 cooling demand	Zone 3 heating demand	Zone 3 cooling demand
l		-0.4634	-0.3198	0.3000	-0.6158	-0.0144	-0.6009	0.1592	0.5517	0.2026	0.5733	0.1544	0.5494
w		-0.5350	-0.4826	0.5252	-0.6767	-0.0553	-0.6532	0.2174	0.6064	0.2237	0.6168	0.2559	0.6359
WWR	S	0.2886	-0.2954	0.2447	0.0094	-0.0200	-0.0175	-0.0137	0.0426	-0.0035	0.0481	-0.0207	0.0425
	N	0.2936	-0.1933	0.2229	0.0243	-0.0266	0.0160	-0.0562	0.0445	-0.0632	0.0490	-0.0601	0.0363
	E	0.3823	-0.2783	0.3410	0.0603	-0.0538	0.0555	0.0810	0.0080	0.0646	-0.0121	0.0690	0.0371
	W	0.3441	-0.2769	0.3517	-0.0168	-0.0836	0.0153	0.0647	0.0157	0.0180	0.0167	0.0377	0.0657
ORI		-0.0721	0.1400	-0.0888	0.0265	0.0071	-0.0002	-0.0575	-0.0202	-0.0560	-0.0235	-0.0609	-0.0304
U_{Wall}		-0.3551	-0.0632	0.0893	-0.1009	0.0456	-0.1166	-0.0260	0.1238	-0.0200	0.1213	-0.0187	0.1079
U_{Win}		-0.0530	-0.5903	0.1160	-0.1122	0.1718	-0.1283	-0.0104	0.1687	0.0591	0.1572	0.0255	0.1458
U_{gfloor}		0.0928	-0.0396	0.0643	-0.1684	0.0713	0.0808	0.1100	-0.1400	0.0019	-0.0331	-0.0006	-0.0205
C_{floor}		-0.0074	0.0623	-0.0994	0.0906	0.1955	0.1408	-0.0124	-0.1084	-0.0403	-0.1070	-0.0382	-0.0962
U_{Roof}		-0.0131	-0.0141	0.0910	0.0170	-0.0885	-0.2394	0.1232	-0.0349	0.1489	-0.0508	0.1781	-0.0732
g_{Win}		-0.1450	-0.0433	0.3850	-0.1193	0.1195	-0.0801	0.0357	0.1295	0.0566	0.1419	0.0493	0.1813
ACH		-0.0025	0.0144	0.0187	-0.0464	-0.6872	-0.0824	0.7344	-0.2931	0.6930	-0.2846	0.7355	-0.2767

also some limitations that need to be discussed in this section.

Firstly, modeling and computation time are drastically reduced, and modeling is simplified compared to a detailed simulation, this being vital for early design phases. At the same time, as is illustrated in Test Case 3, the component-based approach – in contrast to the monolithic ML approaches – provides valuable insight into the processes and flows taking place between the components. This is important knowledge for designers and engineers to permit them to understand what is happening and to improve the design in a goal-oriented way.

The test cases demonstrated the reusability of components in contexts deviating from training structures. Although the test cases are still simple, they show in principle that arbitrary architectural designs can be modeled by components, either at construction element level, or at zone level. The component-based modeling in Test Case 3 particularly illustrates the potential to use component-based modeling to extend covered design space beyond what is possible with parametric black-box models, and to reach a diversity and generality of models similar to that which can be achieved with detailed simulation models. It includes additional components forming a component configuration, thus differing from the configuration in the parametric training model; roof components are added above the lower two levels. Such changes stretch the purely parametric monolithic ML model to its limits, as all possible configurations need to be included in the parametric simulation and in the training data. In contrast, the component-based approach significantly simplifies the parametrics and data generation for the training process, whereas it allows extensive combinatorics in the use of the ML model components. This brings the freedom and simplicity to building performance models that are required for architectural design, especially in early phases.

We assume in practice that it is possible to develop ML components that are sufficiently general to be reused by different designers in their design cases. Designers, planners and engineers are therefore pure users of trained components, and do not need to perform training data generation and ML model training, which is done by tool developers.

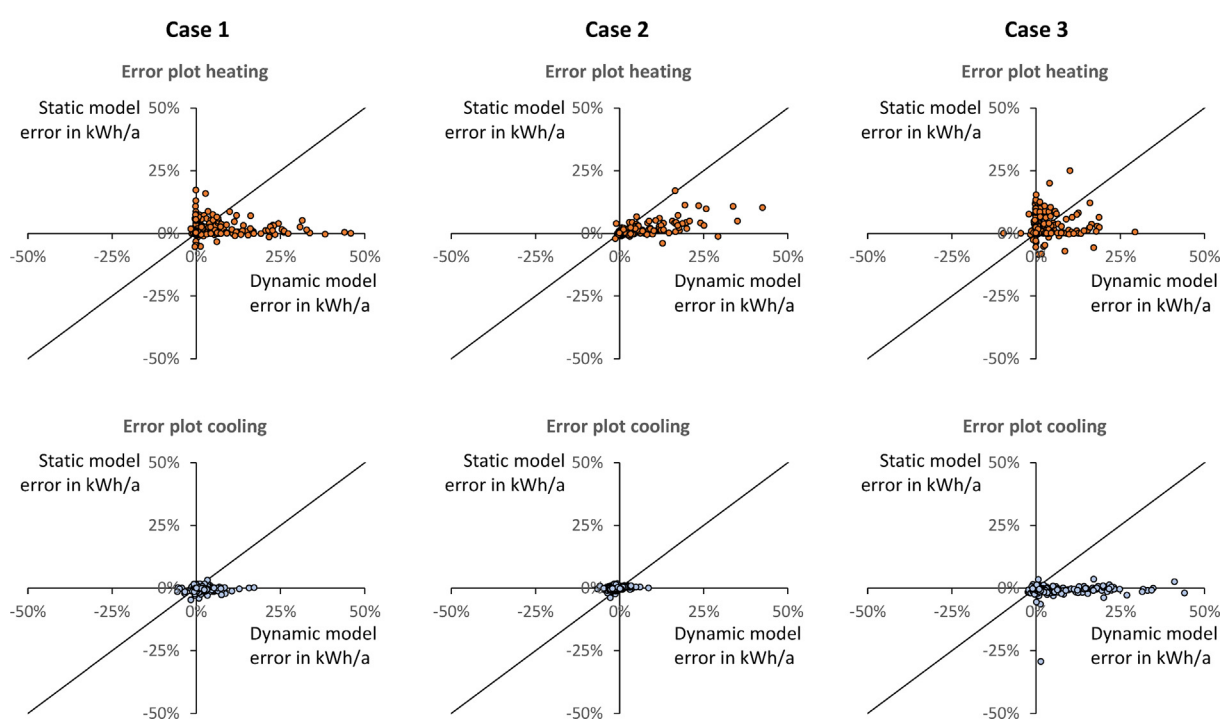


Fig. 18. Error overview in relation to the maximum prediction for all cases and for the static and dynamic models.

Furthermore, component-based ML opens up the option to include data from other sources such as designers' simulations, or to monitor data in such a way that these information-based assets become more generally useable.

Furthermore, the component-based approach represents an engineering-driven method of deep learning. Stacking the component ML model to represent the building model forms a multilayer model, as it is typical of deep learning. In contrast to existing approaches, the component-based approach first uses an engineering structure derived from decomposition, after which additional component-internal or component-crossing layers are then added. The component-based approach nonetheless benefits from the advantages of deep learning, which are the higher abstraction and flexibility of the ML model.

The fundamental limitation of ML is the training data range. Basically, the ML model is only valid for cases that are covered by the training data. However, the abstraction of the component-based approach, following the deep learning paradigm, allows a structural extension of the applicability of the ML model, thus enabling training data to be used: Although the training data is limited to the box-like model, the breakdown into components and their training permits more complex cases to be analyzed, as shown by Test Case 3, and thus extends the design space.

The basic limitation that the ML model is only as good as its training data is still valid. However, as a consequence of the component-based approach, the question of the representation and coverage of the design space by training data shifts from the building level to the component level. One needs to make sure that, in prediction cases, the parameters of the components remain within the range of the training data. Basically, this enables building cases to be predicted that are not included in the training as far as the component ranges have been included in the training, as was shown in the test cases. The last test case particularly shows this by departing from the box-like configuration. The slightly higher inaccuracies in this case are probably caused by the fact that component use is at the limits of the components' training data. Furthermore, we were able to conclude from some of the preliminary tests that the current models, with their training data and parameter settings, are limited to orthogonal cases, since crucial parameters, such as the ratio of the envelope surface to volume, change significantly in the non-orthogonal cases examined. Moreover, self-shading and external shading are not considered in the current models – a feature that requires further research to represent more complex designs.

A further aspect making the component-based approach valuable is BIM integration. Components are aligned to decomposition structures used in BIM. For instance, the wall component with its parameters such as width and window-to-wall ratio can be linked to the wall representation data in BIM. As simple calculations allow the derivation of these parameters, a ML-based energy performance prediction becomes available for BIM models without any complex simulation by the component-based approach. This paves the way for real-time response for performance-based design integrated in BIM.

As part of our future work, we will examine in more detail the dependency of component model validity, depending on sampling and training data. This includes the observation of parameter ranges and the definition of prediction intervals. Furthermore, training data generation and component structure need to be adapted to more complex design cases in order to enable the approach for further architectural designs.

5. Conclusions

This paper has demonstrated a component-based approach of machine learning (ML) for predicting energy performance. ML components have been developed at two levels of decomposition, namely construction-level components with a static prediction of total yearly heating and cooling energy consumption as response, and zone-level components with dynamic monthly consumption as a response. The components' application in test cases that are structurally distinct from the training model demonstrated flexibility and extended generalization beyond the training cases, i.e. the reusability of ML models.

As distinct from the training case, the test cases show how it is possible to generalize beyond the training case with the components. In terms of its parametric structure, the three-storey box building differs from the eight-storey high-rise and the architecturally more complex design in the last test case. These cases demonstrate that the component-based approach enables the flexibility and generalization for ML beyond parametrics, something which is required for the application of ML in building design.

The key to this generalization is that the components typically occur in building design cases, and thus are representative of these cases. The systems engineering-driven decomposition to components represents an engineering-driven deep learning approach in the form of the stacked component models with their embedded ML layers. However, in contrast to conventional deep learning, the abstraction and flexibility are not driven by ML alone, but incorporate engineering knowledge, i.e. structures and engineering considerations in the form of the components given by the domain of building design. This links the deep learning ML approach more closely to engineering. An important feature is interpretable intercomponent results, such as the heat flows resulting from the construction components' prediction, which provides valuable information for engineering interpretation.

In summary, the component-based decomposition of parametric system modeling and the development of general components extends the reusability of the ML model so that they become as generally applicable as components in dynamic performance simulation are today. This allows designers and engineers to use the ML components instead of a physical simulation, with the benefit of drastically-reduced modeling effort and instant feedback in terms of building performance. This allows performance prediction to be directly integrated into the design process of a building, especially in the early design phases. We expect this integration to enable the exploitation of major sustainability potential in building design.

Acknowledgement

The research was funded by Starting Grant STG-14-00346 of the KU Leuven and by the Deutsche Forschungsgemeinschaft (DFG) in Researcher Unit 2363 "Evaluation of building design variants in early phases using adaptive levels of development", in Subproject 4 "System-based Simulation of Energy Flows".

References

[1] Gero JS. Towards a model of exploration in computer-aided design. Gero ETyugu Eds Form. Des. Methods CAD N.-Holl 1994:315–36.
[2] Maher ML, Poon J. Modeling design exploration as co-evolution. Comput-Aided Civ Infrastruct Eng 1996;11:195–209.
[3] Clarke J. Energy simulation in building design. 2nd ed. Oxford: Butterworth-Heinemann; 2001.
[4] Building performance simulation for design and operation. Routledge; 2011.
[5] Trčka M, Hensen J, Wetter M. Co-simulation for performance prediction of integrated building and HVAC systems – an analysis of solution characteristics using a two-body system. Simul Model Pract Theory 2010;18:957–70. https://doi.org/10.1016/j.simpat.2010.02.011.
[6] Welle B, Haymaker J, Rogers Z. ThermalOpt: a methodology for automated BIM-based multidisciplinary thermal simulation for use in optimization environments. Build Simul 2011;4:293–313. https://doi.org/10.1007/s12273-011-0052-5.
[7] Flager F, Welle B, Bansal P, Soremekun G. Multidisciplinary process integration and design optimization of a classroom building. ITcon 2009;14:595–612.
[8] From Zielesny A. Curve fitting to machine learning. Springer Berlin Heidelberg; 2011.
[9] Forrester A, Sóbester A, Keane A. Engineering design via surrogate modelling: a practical guide. Chichester: Wiley; 2008.
[10] Alpaydin E. Introduction to machine learning. The MIT Press; 2010.
[11] Box G, Wilson K. On the experimental attainment of optimum conditions. J R Stat Soc Ser B Methodol 1951;13:1–45.
[12] Box G, Draper N. Response surfaces, mixtures, and ridge analyses. Hoboken, NJ: Wiley; 2007.
[13] Magoulès F, Data Zhao H. Mining and machine learning in building energy analysis. Wiley Online Libr; 2016.

[14] Kramer R, van Schijndel J, Schellen H. Simplified thermal and hygric building models: a literature review. Front Archit Res 2012;1:318–25. https://doi.org/10.1016/j.foar.2012.09.001.
[15] Foucquier A, Robert S, Suard F, Stéphan L. State of the art in building modelling and energy performances prediction: a review. Renew Sustain Energy Rev 2013;23:272–88. https://doi.org/10.1016/j.rser.2013.03.004.
[16] Zhao H, Magoulès F. A review on the prediction of building energy consumption. Renew Sustain Energy Rev 2012;16:3586–92. https://doi.org/10.1016/j.rser.2012.02.049.
[17] Dounis AI. Artificial intelligence for energy conservation in buildings. Adv Build Energy Res 2010;4:267–99. https://doi.org/10.3763/aber.2009.0408.
[18] Chen X, Yang H. A multi-stage optimization of passively designed high-rise residential buildings in multiple building operation scenarios. Appl Energy 2017;206:541–57. https://doi.org/10.1016/j.apenergy.2017.08.204.
[19] Chen X, Yang H, Sun K. Developing a meta-model for sensitivity analyses and prediction of building performance for passively designed high-rise residential buildings. Appl Energy 2017;194:422–39. https://doi.org/10.1016/j.apenergy.2016.08.180.
[20] Stavrakakis G, Zervas P, Sarimveis H, Markatos N. Optimization of window-openings design for thermal comfort in naturally ventilated buildings. Appl Math Model 2012;36:193–211. https://doi.org/10.1016/j.apm.2011.05.052.
[21] Kusiak A, Xu G. Modeling and optimization of HVAC systems using a dynamic neural network. Energy 2012;42:241–50. https://doi.org/10.1016/j.energy.2012.03.063.
[22] Ashtiani A, Mirzaei P, Haghighat F. Indoor thermal condition in urban heat island: comparison of the artificial neural network and regression methods prediction. Energy Build 2014;76:597–604. https://doi.org/10.1016/j.enbuild.2014.03.018.
[23] Jiménez M, Madsen H, Andersen K. Identification of the main thermal characteristics of building components using MATLAB. Build Environ 2008;43:170–80. https://doi.org/10.1016/j.buildenv.2006.10.030.
[24] Yang J, Rivard H, Zmeureanu R. On-line building energy prediction using adaptive artificial neural networks. Energy Build 2005;37:1250–9. https://doi.org/10.1016/j.enbuild.2005.02.005.
[25] Ahmed A, Otreba M, Korres N, Elhadi H. Assessing the performance of naturally day-lit buildings using data mining. Adv Eng Inform 2011;25:364–79. https://doi.org/10.1016/j.aei.2010.09.002.
[26] Kontokosta CE, Tull C. A data-driven predictive model of city-scale energy use in buildings. Appl Energy 2017;197:303–17. https://doi.org/10.1016/j.apenergy.2017.04.005.
[27] Robinson C, Dilkina B, Hubbs J, Zhang W, Guhathakurta S, Brown MA, et al. Machine learning approaches for estimating commercial building energy consumption. Appl Energy 2017;208:889–904. https://doi.org/10.1016/j.apenergy.2017.09.060.
[28] Wei L, Tian W, Silva E, Choudhary R. Comparative study on machine learning for urban building energy analysis. Proc Eng 2015;121:285–92. https://doi.org/10.1016/j.proeng.2015.08.1070.
[29] Aydinalp M, Ismet Ugursal V, Fung AS. Modeling of the space and domestic hot-water heating energy-consumption in the residential sector using neural networks. Appl Energy 2004;79:159–78. https://doi.org/10.1016/j.apenergy.2003.12.006.
[30] Aydinalp-Koksal M, Ugursal V. Comparison of neural network, conditional demand analysis, and engineering approaches for modeling end-use energy consumption in the residential sector. Appl Energy 2008;85:271–96. https://doi.org/10.1016/j.apenergy.2006.09.012.
[31] Neto A, Fiorelli F. Comparison between detailed model simulation and artificial neural network for forecasting building energy consumption. Energy Build 2008;40:2169–76. https://doi.org/10.1016/j.enbuild.2008.06.013.
[32] Ekici B, Aksoy U. Prediction of building energy consumption by using artificial neural networks. Adv Eng Softw 2009;40:356–62. https://doi.org/10.1016/j.advengsoft.2008.05.003.
[33] Ekici B, Aksoy U. Prediction of building energy needs in early stage of design by using ANFIS. Expert Syst Appl 2011;38:5352–8. https://doi.org/10.1016/j.eswa.2010.10.021.
[34] Catalina T, Virgone J, Blanco E. Development and validation of regression models to predict monthly heating demand for residential buildings. Energy Build 2008;40:1825–32. https://doi.org/10.1016/j.enbuild.2008.04.001.
[35] Yalcintas M. Energy-savings predictions for building-equipment retrofits. Energy Build 2008;40:2111–20. https://doi.org/10.1016/j.enbuild.2008.06.008.
[36] Li Q, Meng Q, Cai J, Yoshino H. Applying support vector machine to predict hourly cooling load in the building. Appl Energy 2009;86:2249–56. https://doi.org/10.1016/j.apenergy.2008.11.035.
[37] de Wilde P, Martinez-Ortiz C, Pearson D, Beynon I. Building simulation approaches for the training of automated data analysis tools in building energy management. Adv Eng Inform 2013;27:457–65. https://doi.org/10.1016/j.aei.2013.05.001.
[38] Jain R, Smith K, Culligan P, Taylor J. Forecasting energy consumption of multi-family residential buildings using support vector regression: investigating the impact of temporal and spatial monitoring granularity on performance accuracy. Appl Energy 2014;123:168–78. https://doi.org/10.1016/j.apenergy.2014.02.057.
[39] Guo Y, Wang J, Chen H, Li G, Liu J, Xu C, et al. Machine learning-based thermal response time ahead energy demand prediction for building heating systems. Appl Energy 2018;221:16–27. https://doi.org/10.1016/j.apenergy.2018.03.125.
[40] Chou J-S, Ngo N-T. Time series analytics using sliding window metaheuristic optimization-based machine learning system for identifying building energy consumption patterns. Appl Energy 2016;177:751–70. https://doi.org/10.1016/j.apenergy.2016.05.074.
[41] Yang L, Nagy Z, Goffin P, Schlueter A. Reinforcement learning for optimal control of low exergy buildings. Appl Energy 2015;156:577–86. https://doi.org/10.1016/j.apenergy.2015.07.050.
[42] Ben-Nakhi A, Mahmoud M. Cooling load prediction for buildings using general regression neural networks. Energy Convers Manag 2004;45:2127–41. https://doi.org/10.1016/j.enconman.2003.10.009.
[43] Kalogirou S. Artificial neural networks in energy applications in buildings. Int J Low-Carbon Technol 2006;1:201–16.
[44] Hou Z, Lian Z, Yao Y, Yuan X. Cooling-load prediction by the combination of rough set theory and an artificial neural-network based on data-fusion technique. Appl Energy 2006;83:1033–46. https://doi.org/10.1016/j.apenergy.2005.08.006.
[45] Moon J. Performance of ANN-based predictive and adaptive thermal-control methods for disturbances in and around residential buildings. Build Environ 2012;48:15–26. https://doi.org/10.1016/j.buildenv.2011.06.005.
[46] Idowu S, Saguna S, Åhlund C, Schelén O. Applied machine learning: forecasting heat load in district heating system. Energy Build 2016;133:478–88. https://doi.org/10.1016/j.enbuild.2016.09.068.
[47] Ekren O, Ekren B. Size optimization of a PV/wind hybrid energy conversion system with battery storage using response surface methodology. Appl Energy 2008;85:1086–101. https://doi.org/10.1016/j.apenergy.2008.02.016.
[48] Zhang J, Chowdhury S, Messac A, Castillo L. A response surface-based cost model for wind farm design. Energy Policy 2012;42:538–50. https://doi.org/10.1016/j.enpol.2011.12.021.
[49] Makasis N, Narsilio GA, Bidarmaghz A. A machine learning approach to energy pile design. Comput Geotech 2018;97:189–203. https://doi.org/10.1016/j.compgeo.2018.01.011.
[50] Lee JH, Shin J, Realff MJ. Machine learning: overview of the recent progresses and implications for the process systems engineering field. Comput Chem Eng 2017. https://doi.org/10.1016/j.compchemeng.2017.10.008.
[51] Reynolds J, Ahmad MW, Rezgui Y. Holistic modelling techniques for the operational optimisation of multi-vector energy systems. Energy Build 2018. https://doi.org/10.1016/j.enbuild.2018.03.065.
[52] NASA. Systems Engineering. National Aeronautics and Space Administration; 2007.
[53] Haberfellner R, de Weck O, Fricke E, Vössner S. Systems Engineering – Grundlagen und Anwendung. orell füssli 2012.
[54] Hitchins D. Systems engineering. Chichester: Wiley; 2007.
[55] De Weck O, Roos D, Magee C. Engineering systems: meeting human needs in a complex technological world. Cambridge, Mass. [U.a.]: MIT Press; 2011.
[56] Eppinger S, Browning T. Design structure matrix methods and applications. Cambridge, Mass: MIT Press; 2012.
[57] Browning T. Applying the design structure matrix to system decomposition and integration problems: a review and new directions. IEEE Trans Eng Manag 2001;48:292–306.
[58] Schmidt R, Austin S, Brown D. Designing Adaptable Buildings. 11th Int. Des. Struct. Matrix Conf. DSM'09, 11th International Design Structure Matrix Conference, DSM'09; 2009.
[59] Schmidt III R, Mohyuddin S, Austin S, Gibb A. Using DSM to redefine buildings for adaptability. In: Proc. 10th Int. DSM Conf. Stockh. Swed. 11-12 Novemb. 2008, Proceedings of the 10th International DSM Conference, Stockholm, Sweden, 11-12 November 2008; 2008.
[60] Austin S, Baldwin A, Li B, Waskett P. Analytical design planning technique (ADePT): a dependency structure matrix tool to schedule the building design process. Constr Manag Econ 2000;18:173–82. https://doi.org/10.1080/014461900370807.
[61] Austin S, Baldwin A, Li B, Waskett P. Analytical design planning technique: a model of the detailed building design process. Des Stud 1999;20:279–96.
[62] Geyer P. Systems modelling for sustainable building design. Adv Eng Inform 2012;26:656–68. https://doi.org/10.1016/j.aei.2012.04.005.
[63] Geyer P, Buchholz M. Parametric systems modeling for sustainable energy and resource flows in buildings and their urban environment. Autom Constr 2012;22:70–80. https://doi.org/10.1016/j.autcon.2011.07.002.
[64] Eastman Chuck, Teicholz Paul, Sacks Rafael, Liston Kathleen. BIM handbook. Hoboken, NJ: Wiley; 2011.
[65] Borrmann A, König M, Koch C, Beetz J. Building Information Modeling – Technologische Grundlagen und industrielle Praxis. Springer; 2015.
[66] Shepherd D. BIM management handbook. London: RIBA; 2015.
[67] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521:436–44. https://doi.org/10.1038/nature14539.
[68] Schmidhuber J. Deep learning in neural networks: an overview. Neural Netw 2015;61:85–117. https://doi.org/10.1016/j.neunet.2014.09.003.
[69] Singaravel S, Geyer P, Suykens J. Deep learning neural networks architectures and methods: building design energy prediction by component-based models. Adv Eng Inform 2018;38:81–90.
[70] Object Management Group. Systems Modeling Language, Specifications Version 1.5 2017. < http://www.omg.org/spec/SysML/1.5/ > (accessed June 16, 2017).
[71] Documentation|EnergyPlus n.d. < https://energyplus.net/documentation > (accessed April 5, 2018).
[72] Testing and Validation|EnergyPlus n.d. < https://energyplus.net/testing > (accessed April 5, 2018).
[73] Neymark J, Judkoff R, Beausoleil-Morrison I, Ben-Nakhi A, Crowley M, Deru M, et al. International energy agency building energy simulation test and diagnostic method (IEA BESTEST): in-depth diagnostic cases for ground coupled heat transfer related to slab-on-grade construction. Golden, CO: National Renewable Energy Laboratory (NREL); 2008.
[74] Judkoff R, Polly B, Bianchi M, Neymark J. Building energy simulation test for existing homes (BESTEST-EX) methodology: preprint. Golden, CO: National Renewable Energy Laboratory (NREL); 2011.
[75] Judkoff R, Neymark J. Twenty years on!: updating the IEA BESTEST building thermal fabric test cases for. ASHRAE Standard 140 2013. https://doi.org/10.2172/1220110.
[76] Dogan T, Saratsis E, Reinhart C. The optimization potential of floor-plan typologies in early design energy modeling n.d.