Energy & Buildings 203 (2019) 109405

Contents lists available at ScienceDirect

Energy & Buildings

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

Data-driven modeling of building thermal dynamics: Methodology and state of the art

Zequn Wang*, Yuxiang Chen

Department of Civil and Environmental Engineering, University of Alberta, 9211 116 Street NW, Edmonton, Alberta T6G 1H9, Canada

ARTICLE INFO

Article history:
Received 3 February 2019
Revised 16 August 2019
Accepted 30 August 2019
Available online 31 August 2019

Keywords:
Data-driven models
Building thermal dynamics
RC network
Transfer function
Artificial neural network

ABSTRACT

Data-driven approach is essential to the modeling of building thermal dynamics. It has been widely applied in building operation optimization, energy management, system performance evaluation, and so on. This present paper describes common concepts and fundamental theories of data-driven modeling within the context of building applications. Three types of data-driven models, namely transfer-function (TF) based, resistor-capacitor (RC) based, and artificial-intelligence (AI) based, are critically reviewed, including their formulations, interpretability of physical meanings, and prediction accuracy. Considerations on input and output variables are discussed. Conventional methods and techniques for model training and selection are also presented. Then, the three different models are illustrated through a case study of a real house using on-site monitored data. The case study suggests that the AI model generally outperforms the TF and RC models in predicting indoor temperatures while the RC model is the most appropriate for interpreting the physical behaviours of a building.

1. Introduction

Buildings sector is responsible for over one-third of global final energy use [1]. If no action is taken to improve energy efficiency in the buildings sector, this consumption is expected to rise by 50% by the year 2050 [1]. Besides, almost half the energy consumed in buildings in developed countries is by heating, ventilation, and air conditioning systems [2]. Given this trend and trait of energy demand, a great deal of research efforts has been devoted to enhancing building thermal performance such as operation optimization, energy management, and ongoing commissioning. Among such efforts, data analysis and modeling techniques have demonstrated critical importance [3]. Particularly, data-driven modeling of building thermal dynamics has received growing attention.

The data-driven modeling is also known as an inverse approach, in contrast to an forward approach [4]. The forward approach uses physical principles and prior-known design information (e.g., building geometry and thermal characteristics) to model building responses subjected to boundary conditions. It is mainly used for design. Data-driven models are based on measured data after buildings are occupied. The models reflect the actual building thermal dynamics and give more accurate predictions of building responses. They are often simple, easy to formulate and require less parameterization and computation time. Moreover, multiple model candidates can be easily formulated and compared to derive a more representative model. Therefore, data-driven models can be used in model-based control of space heating and cooling [5–7], fault detection of mechanical systems [8,9], retrofit evaluation [10] for reducing operation energy consumption, shifting and shaving peak demand, and performance monitoring.

Data-driven modeling of building thermal dynamics consists of three phases (see Fig. 1): modeling, training, and selecting. In the modeling phase, a preliminary mathematical model with indefinite parameters is formulated to predict system outputs using measured inputs. For example, a model may predict indoor air temperature using measured weather information, heating or cooling supplies, and occupant activities. The indefinite parameters are then estimated in the training phase by matching the predicted outputs to the measured outputs. Finally, in the selecting phase, the best model is chosen by systematically comparing the well-trained model candidates, mainly by statistical analysis.

The objective of this paper is to conduct a literature review on different data-driven models for building thermal dynamics, along with a demonstration of their applications and analysis of their performances. Section 2 “Categories of Data-driven Models and Their Fundamentals” will demonstrate the concepts, formulations, constraints, and relations of three types of data-driven models. Section 3 “Model Training & Selection” will summarize input/output variables that are commonly employed for model training, and introduce parameter estimation methods, validation criteria, and model selection techniques. In Section 4 “Case Study”,

* Corresponding author.
E-mail address: zequn2@ualberta.ca (Z. Wang).

https://doi.org/10.1016/j.enbuild.2019.109405
0378-7788/

Nomenclature

Acronyms

RC	resistor-capacitor
TF	transfer function
AI	artificial intelligence
ARX	autoregressive with exogenous input
ARMAX	autoregressive moving average with exogenous input
BJ	Box-Jenkins
OE	output error
SS	state space
ANN	artificial neural network
SVM	support vector machine
RNN	recurrent neural network
RBFNN	radial basis function neural network
GRNN	general regression neural network
PEM	prediction error method
MLE	maximum likelihood estimation
MSE	mean square error
RMSE	root mean square error
MAE	mean absolute error
MIMO	multiple-input multiple-output

Variables

u	vector of inputs
y, y	output, the vector of outputs
θ	vector of parameters
T	temperature (°C)
C	thermal capacitance ($kWh/^\circ C$)
R	thermal resistance ($^\circ C/kW$)
F	evaluating factor
Q	the heat input (kW)
x, x	vector of states
$\tilde{A}$, $\tilde{B}$, $\tilde{H}$	matrices in continuous-time state-space representation
A, B, H	matrices in discrete-time state-space representation
K	optimal Kalman gain matrix
e	white Gaussian noise
ω	process noise
v	measurement noise
G, H	transfer functions
$\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, $\mathcal{D}$	polynomial functions
z	forward-shift operator
na, nb, nc, nd	orders of $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, $\mathcal{D}$
nu, ny	number of inputs, number of outputs
ϕ	radiation (kW/m^2)
m	net input signal
f	activation function
nl	number of neurons within the hidden layer
w, w	weight, the vector of weights
b, b	bias, the vector of biases
φ	nonlinear mapping
V	Objective (or loss) function
L	likelihood function
Fit	the goodness of fit
N	number of samples
h	prediction horizon
$\mathbf{\Sigma}$	conditional covariance
ud, yd	input delay, output delay

Subscripts

$\tilde{t}$, t	time, the t^{th} time step
o	outdoor air
s	solar
h	heating (only used with Q)
$elec$	gross electricity
i	indoor air
m	thermal mass
e	building envelop

different models will be compared through their applications on modeling a single thermal zone. Section 5 "Discussion" will discuss major differences between the reviewed models.

2. Categories of data-driven models and their fundamentals

In the literature, there are three main categories of data-driven models for building thermal dynamics, i.e., models based on thermal resistor-capacitor networks (RC models), models based on discrete-time transfer functions (TF models), and models based on artificial intelligence techniques (AI models).

Building thermal process can be nonlinear. For example, surface convective heat transfer is nonlinearly dependent on the surface-air temperature difference. In addition, some thermal properties may vary over time due to changes in environment and operation (e.g., ventilation rate continually changes). Complex model structures can be developed to account for the nonlinearity and time-varying behaviours of building systems [11,12]. However, most researchers adopt linear time-invariant versions of the data-driven models (except for AI models that are naturally nonlinear) for simplicity. Their results have shown that linear time-invariant models can characterize building thermal dynamics with promising performance [12–18].

This study will focus on describing linear time-invariant TF and RC models as well as time-invariant AI models. These models are either for prediction purposes (e.g., predicting room temperatures into the future) or for explanatory purposes (e.g., deriving specific thermal properties). They will be investigated regarding their structure formulations, prediction abilities, and physical interpretations.

2.1. RC models

The RC models capture building thermal dynamics by a network of temperature nodes, thermal resistors, and thermal capacitors (e.g., Fig. 2). Resistances, capacitances, and other necessary parameters in the network are referred to as equivalent thermal parameters [19–21]. Their estimated values enable the RC network to imitate building thermal dynamics, but they do not represent precisely the corresponding apparent quantities. RC models are often used for zone air temperature prediction [13,20–23], as well as heating or cooling load prediction [6,24–27]. In addition, the RC models can also be used for inferring important building thermal characteristics, such as the overall effective thermal transmittance and solar gain factors [28–31]. For example, in Fig. 2, the overall thermal transmittance can be approximated as $U = (R_2 + R_3)^{-1} + {R_1}^{-1}$.

Measured or prior-known values of variables (e.g., zone temperatures) are often called system inputs or outputs, while temperatures linked to thermal capacitors are system states. The number of system states determines a model's order (or the number of ordinary differential equations exemplified in the following context). Typically, models of lower orders are preferable since they require less parameterization and often have satisfactory prediction accuracy compared to larger ones [7,21,29,32–36].

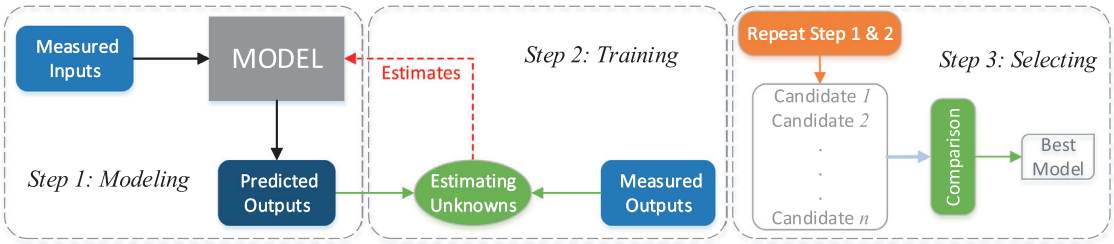


Fig. 1. A general procedure of the data-driven modeling.

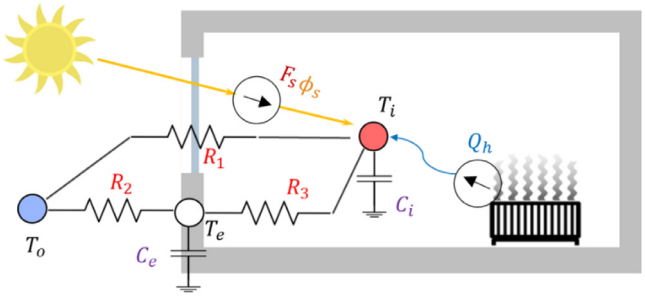


Fig. 2. An example representation of a room with a RC network (T_i: indoor air temperature; T_e: the average temperature of building envelop; T_o: outdoor air temperature; Q_h: heating power; ϕ_s: solar radiation on south façade; F_s: a ratio factor).

Given an RC network and assuming one-dimensional heat transfer, thermal dynamics of each temperature node in the network are governed by the following ordinary differential equation. Using a node numbered k for example (the k can be the node i or e in the RC network in Fig. 2):

$$C_k \frac{dT_k}{dt} = \sum_k \frac{T_k - T_k}{R_{k,k}} + \sum_j F_j Q_j \tag{1}$$

where,

T_k represents the temperature of the k^{th} node;
C_k is the thermal capacitance of the k^{th} node;
T_k represents the temperature of a neighbor of the k^{th} node.
$R_{k,k}$ is the thermal resistance between the k^{th}node and its neighbor;
Q_j denotes the j^{th} heat input to the k^{th} node (e.g., solar radiation); and
F_j is a factor that evaluates effective heat gains to the k^{th} node due to Q_j.

Such ordinary differential equations are set up for all the temperature nodes and rearranged into a state-space representation where model inputs, outputs, states, and parameters are clearly defined.

The model parameters consist of C_k, $R_{k,k}$, and F_j which will be estimated during model training. Table 1 shows the formulation of a simple second-order RC model (Fig. 2), where the model output T_i and the model inputs T_e, ϕ_s, and Q_h are measurable variables.

The state-space representation obtained from differential equations is in continuous-time. For a multiple-input multiple-output (MIMO) system, the continuous-time state-space representation can be written as

$$\frac{dx}{dt} = \tilde{A}x + \tilde{B}u \tag{2a}$$

$$y = \tilde{C}x + \tilde{D}u \tag{2b}$$

where,

$\tilde{t}$ represents time;
x, u, and y are vectors of model states, inputs, and outputs, respectively; and
$\tilde{A}$, $\tilde{B}$, $\tilde{C}$, and $\tilde{D}$ are matrices determined from model parameters e.g., (R_1; R_2; R_3; C_e; C_i; F_s).

Eq. (2) must be converted to a finite difference form in order to make use of the measured data for model training. The conversion of state-space representation is known as discretization and easily accessible in relevant studies [20,37,38]. Two discretization methods are recommended: piecewise constant interpolation and piecewise linear interpolation. They can be found in, e.g., Chapter 6 of [39]. Besides, environmental disturbances and measurement imperfections can result in noise-corrupted data. By taking noise term into account, a discretized state-space representation is expressed as

$$x_{t+1} = Ax_t + Bu_t + Ke_t \tag{3a}$$

$$y_t = Cx_t + Du_t + e_t \tag{3b}$$

where,

t represents the t^{th} time step;
x_t is a vector of states converted from x_t;
A, B, C, and D are matrices calculated from $\tilde{A}$, $\tilde{B}$, $\tilde{C}$, and $\tilde{D}$;
K is the optimal Kalman gain matrix; and
e_t is a vector of stochastic processes, often assumed as normally distributed white noises [15,18,20,40,41].

Eq. (3) is called the innovation form of state-space representation, compared to its stochastic form where two noise terms (process noise ω and measurement noise v) are adopted (i.e., the equations become $x_{t+1} = Ax_t + Bu_t + \omega_t$ and $y_t = Cx_t + Du_t + v_t$). Since the Kalman gain matrix K can be derived from covariances of ω and v through the Algebraic Riccati Equation [42], these two forms of state-space representation are mathematically equivalent [15,43].

2.2. TF models

The outputs of physical (or causal) systems (e.g., indoor air temperature in a building) depend on current and past inputs [44]. Thus, for a linear system, the current output can be related to the history of inputs by the convolution sum: $y_t = \sum_{j=-\infty}^{t} \theta_{t-j} u_j$ or $y_t = (\sum_{j=-\infty}^{t} \theta_{t-j} z^{-j}) u_t$, where θ's are constant parameters and z is a forward-shift operator: $u_{t-1} = z^{-1} u_t$. Here, $\sum_{j=-\infty}^{t} \theta_{t-j} z^{-j}$ can be regarded as a transfer function (TF) in discrete time. Such TFs can be used to characterize the input-output relationships and thermal dynamics of buildings. In general, a linear time-invariant TF model can be written as [44]:

$$y_t = G(\theta, z) u_t + H(\theta, z) e_t \tag{4}$$

where,

Table 1
An example of RC model formulation.

Model structure	RC network in Fig. 2
Variables	Input: $u = [T_o \quad \phi_s \quad Q_h]'$; Output: $y = T_i$; State: $x = [T_e \quad T_i]'$; Parameter: $\theta = [R_1 \quad R_2 \quad R_3 \quad C_e \quad C_i \quad F_s \quad F_h]'$
Ordinary differential equations	$C_e \frac{dT_e}{dt} = \frac{T_o - T_e}{R_2} + \frac{T_i - T_e}{R_3}$ $C_i \frac{dT_i}{dt} = \frac{T_o - T_i}{R_1} + \frac{T_e - T_i}{R_3} + F_s\phi_s + F_hQ_h \ (F_h = 1)$
State-space representation in continuous-time	$\frac{d}{dt}\begin{bmatrix} T_e \\ T_i \end{bmatrix} = \underbrace{\begin{bmatrix} -\frac{1}{C_eR_2} - \frac{1}{C_eR_3} & \frac{1}{C_eR_3} \\ \frac{1}{C_iR_3} & -\frac{1}{C_iR_1} - \frac{1}{C_iR_3} \end{bmatrix}}_{\tilde{A}} \begin{bmatrix} T_e \\ T_i \end{bmatrix} + \underbrace{\begin{bmatrix} \frac{1}{C_eR_2} & 0 & 0 \\ \frac{1}{C_iR_1} & \frac{F_s}{C_i} & \frac{1}{C_i} \end{bmatrix}}_{\tilde{B}} \begin{bmatrix} T_o \\ \phi_s \\ Q_h \end{bmatrix}$ $T_i = \underbrace{[0 \quad 1]}_{\tilde{C}} \begin{bmatrix} T_e \\ T_i \end{bmatrix} + \underbrace{0 \quad 0 \quad 0}_{\tilde{D}} \begin{bmatrix} T_o \\ \phi_s \\ Q_h \end{bmatrix}$

z is a forward-shift operator: $u_{t+1} = zu_t$ or $u_t = z^{-1}u_{t+1}$;
$G(\theta, z)$ is the transfer function for inputs;
$H(\theta, z)$ is the transfer function for system noise; and
e_t is the noise term accounting for system disturbances.

$G(\theta, z)$ and $H(\theta, z)$ are often expressed in rational function forms, e.g., $G(\theta, z) = \mathcal{B}(z)/\mathcal{A}(z)$ and $H(\theta, z) = \mathcal{C}(z)/\mathcal{D}(z)$, where $\mathcal{A}(z)$, $\mathcal{B}(z)$, $\mathcal{C}(z)$, and $\mathcal{D}(z)$ are polynomial functions:

$$\mathcal{A}(z) = a_0 + a_1z^{-1} + a_2z^{-2} + \ldots + a_{na}z^{-na}$$
$$\mathcal{B}(z) = b_0 + b_1z^{-1} + b_2z^{-2} + \ldots + b_{nb}z^{-nb}$$
$$\mathcal{C}(z) = c_0 + c_1z^{-1} + c_2z^{-2} + \ldots + c_{nc}z^{-nc}$$
$$\mathcal{D}(z) = d_0 + d_1z^{-1} + d_2z^{-2} + \ldots + d_{nd}z^{-nd}$$

The vector of model parameters θ is composed of a_j, b_j, c_j, and d_j. The model order is defined by *na, nb, nc*, and *nd*. For a MIMO system with *nu* inputs and *ny* outputs, $G(\theta, z)$ and $H(\theta, z)$ are $ny \times nu$ and $ny \times ny$ matrices of transfer functions, respectively.

Depending on the use of polynomial functions, TF models can be categorized into Box-Jenkins (BJ) model, autoregressive moving average with exogenous input (ARMAX) model, autoregressive with exogenous input (ARX) model, output error (OE) model, etc. [45,46]. Their corresponding mathematical expressions are:

BJ	$y_t = \frac{\mathcal{B}(z)}{\mathcal{A}(z)}u_t + \frac{\mathcal{C}(z)}{\mathcal{D}(z)}e_t$
ARMAX	$y_t = \frac{\mathcal{B}(z)}{\mathcal{A}(z)}u_t + \frac{\mathcal{C}(z)}{\mathcal{A}(z)}e_t$
ARX	$y_t = \frac{\mathcal{B}(z)}{\mathcal{A}(z)}u_t + \frac{1}{\mathcal{A}(z)}e_t$
OE	$y_t = \frac{\mathcal{B}(z)}{\mathcal{A}(z)}u_t + e_t$

These models are different merely by the structure of $H(\theta, z)$, namely, how the noise is modelled. An $H(\theta, z)$ with more freedom (such as that in a BJ model) allows more dynamics from the noise term (e_t) to be described. Hence, the BJ models may outperform the ARMAX and ARX models in terms of prediction accuracy [45]. However, the ARMAX and ARX models are the most popular among researchers because they are simple and suitable for building applications [12,47–54]. Furthermore, the ARMAX or ARX model can be written in a polynomial form (i.e., $\mathcal{A}(z)y_t = \mathcal{B}(z)u_t + \mathcal{C}(z)e_t$ or $\mathcal{A}(z)y_t = \mathcal{B}(z)u_t + e_t$, $a_0 = 1$). The deterministic part of their polynomial forms (i.e., $\mathcal{A}(z)y_t = \mathcal{B}(z)u_t$) has already been derived in the forward approach, known as the comprehensive room transfer function [55]. Therefore, such models in nature, tend to align with the fundamental physical laws.

Other than the abovementioned categories of TF models, another one is the state-space (SS) models. SS models have the same mathematical expression as in Eq. (3). But instead of using equivalent thermal parameters, the matrices *A, B, C, D* and *K* in SS models are directly parameterized with constant entries. For example, in the state-space representation of RC models: $A = [\begin{matrix} -\frac{1}{C_eR_2} - \frac{1}{C_eR_3} & \frac{1}{C_eR_3} \\ \frac{1}{C_iR_3} & -\frac{1}{C_iR_1} - \frac{1}{C_iR_3} \end{matrix}]$; while in SS models: $A = [\begin{matrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{matrix}]$, where a_{11}, a_{12}, a_{21}, and a_{22} are constant parameters to be estimated.

An identified SS model can always be transformed to a unique TF model [56]. By applying $x_{t+1} = zx_t$ to Eq. (3a), we have $(zI - A)x_t = Bu_t + Ke_t$ where I is an identity matrix. Substitute this to Eq. (3b), we get

$$y_t = \left[C(zI - A)^{-1}B + D\right]u_t + \left[C(zI - A)^{-1}K + I\right]e_t \tag{5}$$

In Eq. (5), $[C(zI - A)^{-1}B + D]$ and $[C(zI - A)^{-1}K + I]$ can be explained as transfer functions similar to $G(\theta, z)$ and $H(\theta, z)$. After being transformed to transfer function forms, the SS models have similar structures as ARX models and are also commonly used for characterizing building thermal dynamics [15,18,43,57,58]. Eq. (5) also indicates that RC and TF models are closely related through the state-space representation. However, it is practically impossible to derive all equivalent thermal parameters (e.g., R_3 and C_e) from the SS model's matrix entries (e.g. a_{11}, a_{12}, a_{21}, and a_{22}).

Table 2 gives an example of TF model formulation using ARX structure. It can be seen that the parameters in this ARX model are not as physically interpretable as in the RC model given in Table 1. However, a TF model is more straightforward to develop, i.e., there is no need for an in-depth understanding of thermal systems or solving continuous-time differential equations. As such, the obtained TF models are mostly used for prediction purposes: predicting room temperatures [58,59], humidity level [12,45,60], and heating or cooling load [11,61–63].

Furthermore, TF models are pure statistical models. Being identified from data that contains disturbances, they may violate conservation of energy, exhibit resonant behavior, or be unstable or non-casual (e.g., the simulation of indoor air temperature grows without bound) [49]. Therefore, certain constraints (e.g., steady-state [49,64,65] and pole constraints [49]) or supervisory rules [66] should be applied to the models. The purpose of constraints is to obtain physically plausible and stable models. If the constraints are satisfied, a TF model may also be used for deriving thermal properties of the building, such as the thermal transmittance of building envelop, solar aperture, and time constants [15,46,65].

2.3. AI models

Artificial intelligence (AI) is "the science and engineering of making intelligent machines, especially intelligent computer programs" as defined by John McCarthy [67]. AI techniques (e.g., artificial neural networks) have been extensively adopted for building energy use prediction since the 1990s [68–70]. A representative

Table 2
An example of TF model (ARX) formulation.

Model structure	ARX (na, nb) $nb = [nb_1 \quad nb_2 \quad nb_3]$
Variables	Input: $u = [T_o \quad \phi_s \quad Q_h]'$; Output: $y = T_i$; Parameter: $\theta = [a_1 \quad b_{1,0} \quad b_{1,1} \quad b_{2,0} \quad b_{2,1} \quad b_{3,0}]'$
Transfer function form	$(1 + a_1 z^{-1} + \ldots + a_{na} z^{-na}) T_{i,t} = \begin{bmatrix} b_{1,0} + b_{1,1} z^{-1} + \ldots + b_{1,nb_1} z^{-nb_1} \\ b_{2,0} + b_{2,1} z^{-1} + \ldots + b_{2,nb_2} z^{-nb_2} \\ b_{3,0} + b_{3,1} z^{-1} + \ldots + b_{3,nb_3} z^{-nb_3} \end{bmatrix} [T_{o,t} \quad \phi_{s,t} \quad Q_{h,t}] + e_t$
Polynomial form	e.g., $na = 1$, $nb = [2 \quad 2 \quad 1]$ $T_{i,t} = -a_1 T_{1,t-1} + b_{1,0} T_{o,t} + b_{1,1} T_{o,t-1} + b_{2,0} \phi_{s,t} + b_{2,1} \phi_{s,t-1} + b_{3,0} Q_{h,t} + e_t$

example is the "Great Energy Predictor Shootout I & II" competitions (in 1993 and 1995, respectively) held by American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE), where artificial neural networks provided the most accurate predictions of a building's energy use [71,72]. Among a variety of machine learning techniques, Artificial Neural Networks (ANN) and Support Vector Machines (SVM) are most commonly used for modeling building thermal dynamics [73–75]. Other AI based modeling methods are also adopted, such as artificial immune systems [76] and random forests [77], but they have received relatively less attention than ANN and SVM models do.

Analogous to human brains, ANN models relate inputs and outputs by creating a large structure of artificial neurons [78]. Typical ANN models for building thermal dynamics contain an input layer, one hidden layer, and an output layer [69,70,79,80]. Table 3 shows a simple AI model structured by a three-layered ANN, where one of the artificial neurons is highlighted. In each neuron, the input signals are weighted, summed and added by a bias. Then the net input signal m_t^j is assigned to an activation function $f(\cdot)$ and the activation is passed to the output function (often a linear function) to calculate the output. The example in Table 3 is a feedforward neural network, i.e., the input signals flow forward with no feedbacks. Its counterpart is a recurrent neural network (RNN), where the information can travel in loops from layer to layer [69,81–83]. For example, in Table 3, if $T_{i,\,t}$ (which is measured) in the input layer is replaced by the model prediction $\hat{T}_{i,t}$ from the output layer, the feedforward neural network will turn into an RNN.

The choice of activation function has a major affect on the ANN's performance. In general, the activation function introduces a degree of nonlinearity that is valuable for most ANNs [84], but there is no established rule defined for selecting activation functions to produce better network outputs [73]. Common activation functions include logistic-sigmoid functions [83,85], hyperbolic tangent functions [69,79,80,86], radial basis functions [87–90], etc. Specifically, ANNs that use radial basis functions are referred to as radial basis function neural networks (RBFNN). Such networks are said to have fast online learning ability, strong tolerance to noisy input data, good generalization, and easy design implementation [89]. A variation of RBFNN is the general regression neural network (GRNN), which also uses radial basis functions for activation and is even more suitable for online identification [91,92].

The ANN model's performance is also impacted by the number of neurons within the hidden layer (nl). Too many neurons will cause the network to be overfitted and not generalize well beyond the training data. Too few neurons will weaken the network's ability to learn from the measurements. However, there is no strict rule for determining the right number of hidden neurons. Some researchers use empirical equations to calculate nl from the number of inputs nu and/or the number of outputs ny [83,85,93,94], for example, $nl = 2nu + 1$. Other researchers consider nl as an indicator of model complexity and reduce it during model training or selecting without jeopardizing the prediction accuracy [10,60,79,87]. Model training and selection will be further discussed in Section 3.

In addition to ANN, Support Vector Machines (SVM) are also used increasingly in the modeling of building thermal dynamics [95–97]. The basic idea behind SVM is to map the input space into a high dimensional feature space through some nonlinear mapping, and then perform a linear regression in this feature space [98], namely,

$$\hat{y} = w \cdot \varphi(u) + b \tag{6}$$

where,

$\hat{y}$ represents the predicted output vector;
$\varphi(\cdot)$ denotes the nonlinear mapping;
w is a weight vector; and$b$ is a bias vector.

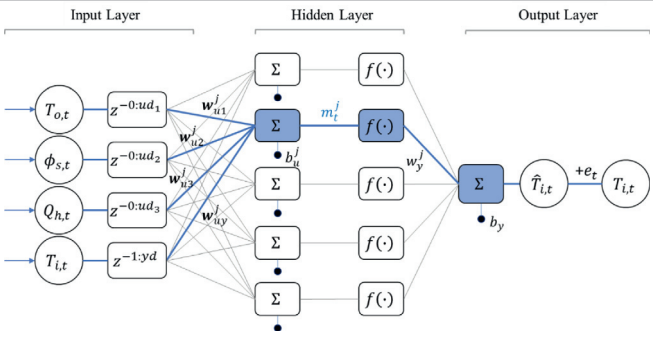
The nonlinear mapping in Eq. (6) extracts nonlinear features from inputs. For example, when $u = [T_o, \ \phi_s, Q_h]$, the mapping can be $\varphi(u) = [T_o, \ \phi_s, Q_h, T_o\phi_s, {Q_h}^2]$. Thus, a three-dimensional input space is mapped into a five-dimensional input space by including two nonlinear features: $T_o\phi_s$ and ${Q_h}^2$. In practice, the realization of $\varphi(\cdot)$ is often implicitly defined via kernels [95,99]. By using kernels, all necessary computations can be performed directly in the input space u without having to compute the mapping $\varphi(\cdot)$ [100]. A detailed description of applying kernels in SVMs can be found in e.g., [101]. Furthermore, a major advantage of the SVM models is that they employ the structural risk minimization principle, which seeks to minimize an upper bound of the generalization error consisting of the sum of the training error and a confidence level [90,96,97,100]. Owing to this advantage, the SVM models can have fewer free parameters and achieve better accuracy and generalization than conventional ANN and RBFNN models in e.g., predicting the hourly cooling load of buildings [90].

For thermal dynamic problems, AI models (ANN and SVM models in particular) can be regarded as nonlinear regressions, where the system inputs are regressors, and the outputs are dependent variables. In general, AI models nonlinearly and implicitly relates outputs to inputs. They can also account for complex interactions between inputs, such as using an intertwined network in the input layer in ANN models or a nonlinear mapping of the input space in SVM models. Unlike RC or some TF models, AI models are not physically interpretable, hence cannot serve for explanatory purposes. However, they tend to have higher prediction accuracy than the linear models [60,70]. For that reason, they have also been extensively applied just like the RC or TF models, in model predictive control [88], fault detection [92,94], retrofit evaluation [10], etc.

2.4. Enhanced models

RC, TF, and AI models can be enhanced by creating hybrid models. One possibility is to combine several different models by assigning linear weights to their outputs, so the combined model can possibly take advantage of each individual model and gives higher prediction accuracy [99,102,103]. Another possibility is to combine a model with techniques such as fuzzy logic [104,105] and wavelet transform [105,106] to improve the model's prediction performance. Furthermore, the basic models can be modified with respect to weather conditions (e.g., outdoor air temperature) or system changes (e.g., opening of windows) to create a series of weather or system dependent models [61,107]. This is equivalent to vary a model's structure or parameters with weather conditions

Table 3
An example of AI model (ANN) formulation.

Model Structure	
Variables	Input: $\boldsymbol{u} = [T_o \quad \phi_s \quad Q_h]'$; Output: $\boldsymbol{y} = T_i$; Parameter: $\boldsymbol{\theta} = \{\boldsymbol{w}_{u1}^{1:nl} \quad \boldsymbol{w}_{u2}^{1:nl} \quad \boldsymbol{w}_{u3}^{1:nl} \quad \boldsymbol{w}_{uy}^{1:nl} \quad b_u^{1:nl} \quad w_y^{1:nl} \quad b_y\}$; e.g., $w_y^{1:nl} = [w_y^1, w_y^2, \dots, w_y^{nl}]$
Net input signal	For the j^{th} neuron: $m_t^j = z^{-1:yd} T_{i,t} \boldsymbol{w}_{uy}^j + z^{-0:ud_1} T_{o,t} \boldsymbol{w}_{u1}^j + z^{-0:ud_2} \phi_{s,t} \boldsymbol{w}_{u2}^j + z^{-0:ud_3} Q_{h,t} \boldsymbol{w}_{u3}^j + b_u^j$ where, nl is the number of neurons ($nl = 5$ in this example); ud_1, ud_2, and ud_3 are input delays (e.g. $z^{-0:ud_1} = [1, z^{-1}, z^{-2}, \dots, z^{-ud_1}]$); yd is output delay ($z^{-1:yd} = [z^{-1}, z^{-2}, \dots, z^{-yd}]$);
	$\boldsymbol{w}_{uy}^j$, $\boldsymbol{w}_{u1}^j$, $\boldsymbol{w}_{u2}^j$, and $\boldsymbol{w}_{u3}^j$ are weight vectors for the j^{th} neuron in the input layer; b_u^j is a bias for the j^{th} neuron in the input layer; w_y^j is a weight for the j^{th} neuron in the output layer; and b_y is a bias in the output layer.
Activation & output functions	$T_{i,t} = \underbrace{\sum_j w_y^j f(m_t^j) + b_y}_{\hat{T}_{i,t}} + e_t$ e.g., $f(m_t^j) = \tanh(m_t^j)$

and system changes. Its goal is to enhance the original model's ability to learn from the available data.

3. Model training and selection

The data-driven models are formulated with indefinite parameters. When training the models to learn buildings' thermal behaviours from the measured data, the indefinite parameters are estimated. After being trained, the models are evaluated by performing residual analysis or testing their generalization on new datasets. Then, the evaluation results are used as an indicator for comparing different models in the selecting phase. This section will focus on commonly adopted methods for training and selection of data-driven models along with typical input and output variables in building thermal dynamic studies.

3.1. Input/output variables

Inputs and outputs to be used for data-driven models are measured as time series. This time series is usually sampled at a fixed sampling interval over a long duration (varies from several weeks to several years depending on the specific model and the data quality). The outputs should be easily observable responses, and the inputs should have significant influences on the system. Generally, input and output variables required for a thermal zone fall into the following types: zone air temperature, outdoor air

temperature, solar radiation, heating or cooling power supply, and internal gains.

For solar gains, some researchers directly use global solar radiation as an input [29,59,108]. Other researchers employ mathematical methods to split the global solar radiation into direct normal and diffuse components to account for their distinctive solar effects [11,63]. It is also possible to combine solar radiation and outdoor air temperature to a single input, i.e., sol-air temperature [26,49,50]. Heating or cooling power supply, in general, is derived from other measured variables depending on the type of heating or cooling system. For examples, in a forced-air heating system, the heating power can be approximated using the measured flow rate and temperature of the supply air from air handling units [6,45]. Internal gains consist of heat gains from occupants, lighting, appliances, equipment, etc. For residential houses, some researchers use constant periodic values to approximate internal heat gains instead of measuring them [11,12,63]. For commercial buildings, many researchers relate internal gains to gross electricity demand excluding that for heating or cooling [15,32,108]. This is because occupants' behaviours can be highly correlated to the equipment (e.g., lights, printers, and computers) being used [109].

Other variables to be measured include ground temperature, wind speed, wind direction, relative humidity, mechanical ventilation, etc. Since the ground temperature varies negligibly compared to other inputs, it is often assumed to be constant [13,15]. Wind speed and direction can change over time the conductive or convective heat transfer coefficient associated with building envelop [11,29,63]. Outdoor relative humidity can also influence zone air temperatures and heating and cooling loads through the humidification or dehumidification process. However, it is indicated that the wind effects as well as the humidity level are not as relevant as the ambient temperature and solar radiation [12,79,87]. In mechanical ventilation, the fresh air intake can be considered as a heat loss to (or a heat gain from) the outdoor environment, which can be treated as an offset of the heating or cooling power supply [29].

Measured variables are prepared for model training and selection. It is generally necessary to acquire input and output data with large and persistent variations in order to obtain reliable parameter estimates and accurate models that can provide accurate predictions [110]. If building systems are not significantly excited, the data will not be as dynamically informative as needed for robust parameter estimation, e.g., certain parameters could be non-identifiable (cannot be uniquely identified) [111]. Therefore, ensuring the quality of on-site measurements is as equally important as developing a good model structure.

3.2. Model training

Model training is an optimization process that estimates the unknown parameters by minimizing the value of an objective function. If the parameter estimate is denoted by $\hat{\theta}$, then

$$\hat{\theta} = \min_{\theta} V(\theta) \tag{7}$$

$V(\cdot)$ in Eq. (7) is a parameter-dependent objective function (or loss function) often defined by prediction error method or maximum likelihood estimation method.

3.2.1. Prediction error method (PEM)

In PEM, the loss function for minimization is a function of prediction errors (usually defined in the quadratic form) [112], for example,

$$V(\theta) = \frac{1}{N}\sum_{t=1}^{N} e_t(\theta)' e_t(\theta) \tag{8}$$

where,

N is the number of samples;
$e_t(\theta)$ is the prediction error at time t: $e_t(\theta) = y_t - \hat{y}_{t|t-h}(\theta)$;
$e_t(\theta)'$ is the transpose of $e_t(\theta)$;
h is the prediction horizon;
$\hat{y}_{t|t-h}(\theta)$ is the predicted output at t given θ and Y_{t-h}; and
Y_{t-h} contains the measured outputs up to $t-h$: $Y_{t-h} = [y_{t-h},\ y_{t-h-1},\ \ldots,\ y_1]$.

When $h = 1$, the error is referred to as one-step ahead prediction error, and the model parameters that give the smallest loss function value will give the best one-step ahead prediction performance. When $h > 1$, the error becomes multi-step ahead prediction error. A loss function with $h > 1$ is especially suitable for model predictive controllers since they often require models that can provide good predictions over a finite-time horizon [113–117]. When $h = \infty$, the error is known as simulation error. This corresponds to, e.g., $H(\theta, z) = 1$ for TF models in Eq. (4). Such TF models are so-called output error (OE) models whose identification has been a longstanding topic studied in various respects [118–120]. By taking $H(\theta, z) = 1$, the OE models focus on the dynamics of inputs and not the disturbance properties of noise. In sum, the prediction horizon influences the estimation results. Choice of h should be based on the purpose: whether the model is for short-term or long-term prediction.

3.2.2. Maximum likelihood estimation (MLE)

The MLE method finds parameter estimates that make the observations (i.e. the measured outputs) most likely to be within the predicted outputs. In other words, the joint probability density of all the observations - the likelihood function - should be maximized. Maximizing the likelihood function is to minimize the loss function:

$$V(\theta) = -\log L(\theta;\ Y_N) \tag{9}$$

where,

$L(\cdot)$ is the likelihood function: $L(\theta;\ Y_N) = \prod_{t=1}^{N} p(y_t|Y_{t-1}, \theta)$;
Y_t contains the observations up to time t: $Y_t = [y_t,\ y_{t-1},\ \ldots,\ y_1]$; and
$p(\cdot|\cdot)$ is the conditional density function, often assumed to be Gaussian.

The Gaussian densities are determined by the conditional mean $\hat{y}_{t|t-1}$ and the conditional covariance $\mathbf{\Sigma}_{t|t-1}$, i.e.,

$$p(y_t|Y_{t-1}, \theta) = \exp\left[-\frac{1}{2}\left(y_t - \hat{y}_{t|t-1}\right)' \mathbf{\Sigma}_{t|t-1}\left(y_t - \hat{y}_{t|t-1}\right)\right] / \sqrt{(2\pi)^{ny}\left|\mathbf{\Sigma}_{t|t-1}\right|}$$

In the stochastic state-space representation, $\hat{y}_{t|t-1}$ and $\mathbf{\Sigma}_{t|t-1}$ can be calculated recursively by using a Kalman filter [20,121]. If the conditional covariance is assumed to be a time -invariant constant (i.e., $\mathbf{\Sigma}_{t|t-1} = \mathbf{\Sigma}$), it can be found by minimizing the loss function with respect to $\mathbf{\Sigma}$ independently from the other parameters [112,122]. Under the special conditions of Gaussian densities and $\mathbf{\Sigma}$, the MLE method is equivalent to the one-step ahead PEM in its least-squares form [122]. As such, a major advantage of the PEM is that no probabilistic assumptions must be made.

3.2.3. Optimization algorithms

Either PEM or MLE defines a loss function to be minimized. For specific models, the value of the loss function can be minimized by linear regression techniques, e.g., linear least squares for the ARX models [123,124] and subspace identification for the state space (SS) models [125,126]. When linear regression techniques are allowed, a global minimum is always guaranteed.

For other models in general, this minimization process can be realized by iterative search algorithms such as Gauss-Newton algorithm [16,32] and Levenberg-Marquardt algorithm

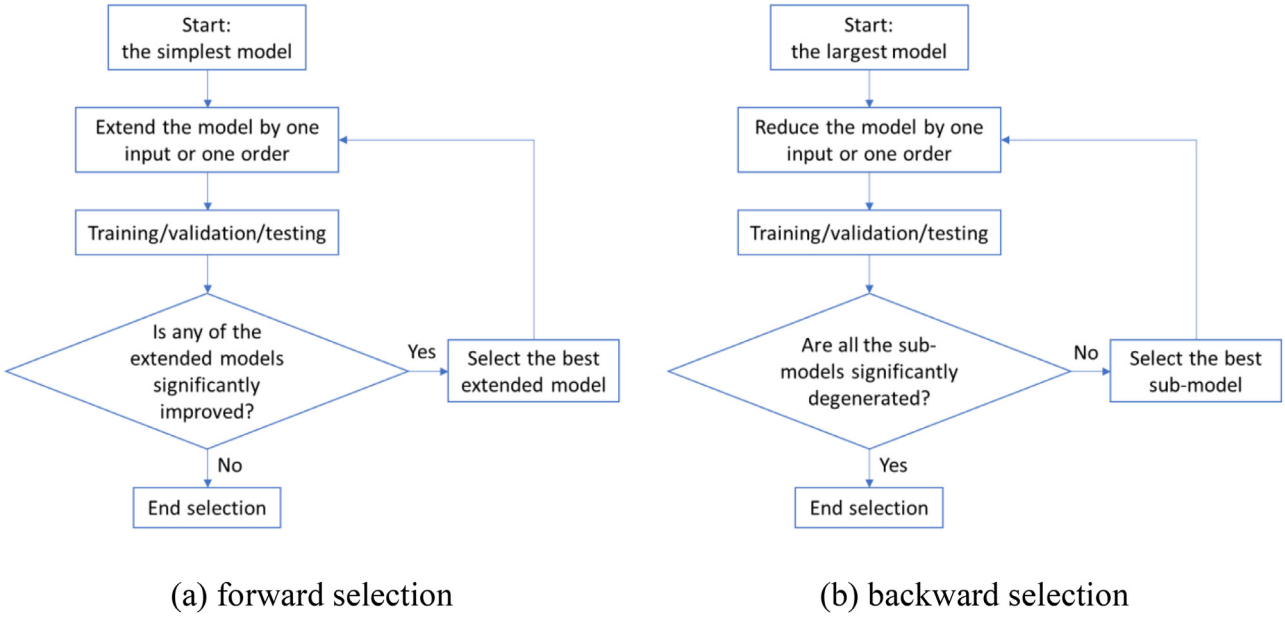


Fig. 3. Model selection procedures: (a) forward, and (b) backward.

[15,24,60,87,88,94]. Typically, these algorithms evaluate gradients or Hessians for a searching direction that leads to the optimal solution. Specifically, backpropagation is a method used in ANN models for calculating gradients [10,60,69,83]. However, the iterative algorithms require a proper initial guess for the parameter estimation. Without a good starting point, the searching guided by gradients or Hessians may lead to a local minimum. Therefore, some authors have proposed to employ global optimization routines, such as the genetic algorithm [26], the multi-start searching [24], the modal trimming method [80], the differential evolution algorithm [97], etc., to approach the global optimal.

3.2.4. On-line algorithms

The above presented algorithms are intended for model training with static datasets (i.e., the size of the dataset is fixed before the algorithms start to optimize the parameter values). Such model training algorithms are known as off-line algorithms. Their counterparts are on-line algorithms, which update the parameter estimates continuously as new data become available for upcoming predictions [33,52,53]. A common practice for updating the estimates is to use a forgetting factor which introduces increasingly weaker weighting on the older data [127]. In other words, the on-line algorithm gradually "forgets" past data as the parameters are recursively estimated. Thus, the parameter estimation using on-line algorithms is also referred to as real-time estimation [53]. Since the estimation of parameters is updated recursively the definition of "real-time" should be distinguished from "adaptive". For example, accumulative training or sliding window training [85,87] adapts to new data but uses off-line (non-recursive) algorithms.

3.3. Model selection

Models trained by off-line algorithms are candidates for model selection, a systematic routine through which these candidates are validated, tested, and compared. The purpose of model selection is to find the most suitable model structure that gives favourable prediction accuracy (and physical interpretations in the cases of RC and TF models) with the least possible complexity. In other words, the selected model should be able to characterize the principal thermal dynamics using least input variables and model parameters.

3.3.1. Forward/backward selection

Two types of model selection are shown in Fig. 3. One is forward selection which starts from the simplest model and progressively extends the model until the model's performance can no longer be improved in a significant sense [15,17,41,63,128]. The other is backward selection that consecutively creates sub-models from a large model until a decrease of the model's performance allows for no further model reduction [86,103,129]. To perform either selection process, the candidate models' performances must be evaluated by validation or/and testing (here, *validation* refers to any analyses conducted on the dataset used for training while *testing* refers to any analyses conducted on an independent dataset from training).

3.3.2. Model evaluation/comparison

The trained models can be evaluated by quality criteria including mean square error (MSE), root mean square error (RMSE), mean absolute error (MAE), the goodness of fit (*Fit*) (see Eq. (11)), the coefficient of determination (R^2), the coefficient of variation (CV), mean bias error, and so on [17,40,45,59,60,79,86,103,128,129]. Choosing which criterion to use depends on how the residuals (i.e., after-training errors) should be penalized and interpreted. For examples, the MSE takes the average of the squared residuals and emphasizes larger errors; the *Fit* is expressed in percentages and suitable to assess how much variance of the output variable is explained by the model. These quality criteria can be used for either validating a model's training performance or testing its generalization (reproductivity) on new observations. Essentially, they measure a model's success at fitting the available data.

Another category of methods for evaluating the trained models is to perform residual analyses based on autocorrelation functions, cumulative periodograms, and cross-correlation functions [15,17,29,41,63,128]. They are used to check for the whiteness or independence of the residuals. Given a pre-specified level of significance, if the hypothesis of whiteness or independence is rejected, there could be significant dynamics remaining unexplained in the residuals, which indicates the need for a larger model or more input variables.

Table 4
Summary of input and output variables.

Variables		Unit	Description
Input	T_o	°C	Outdoor air temperature
	ϕ_s	kW/m^2	Global irradiation on the south façade
	Q_h	kW	Heating power provided by the geothermal heat pump
	Q_{elec}	kW	Gross electricity demand
Output	T_i	°C	Indoor air temperature (average)

Additionally, the likelihood ratio test is a useful tool for comparing two nested models, i.e., one model is the sub-model of the other [17,41]. If the likelihood of the sub-model is significantly less than the larger model, it indicates that reducing the larger model can cause notable decrease in the model performance. Frequency response analysis can also be used for model comparison [21,130]. The basic idea is that for two models of different orders, the one of lower order can be used if their frequency responses resemble each other, which leads to statistically negligible difference between their outputs.

3.3.3. Model pruning

Finally, excessive model comparisons can be caused when there are a lot of possibilities of reducing or extending a model. This is true for complex models (e.g., ANN models) with large sets of input variables and model parameters. To make the selection more efficient, techniques like model pruning [60,79,86] and recursive feature elimination [103,129] can be integrated into the backward selection procedure. Such techniques identify unnecessary parameters or unimportant input variables right after model training. Then the unnecessary parameters or unimportant inputs are removed until a decrease of the prediction accuracy is no longer tolerated. Model pruning or recursive feature elimination can effectively guide model reduction. Moreover, model pruning is a solution to the potential overfitting problem of the ANN models.

4. Case study

This section illustrates the data-driven modeling through a case study of a single-detached house [131,132]. The house has large glazing areas facing south and a significant amount of thermal mass from concrete floor/walls. Its wood-frame building envelop is well-insulated and air-tight. Space heating is mainly provided by a geothermal heat pump through forced hot air.

4.1. Measured data

The house is simplified as one thermal zone. Relevant input and output variables are summarized in Table 4. Outdoor air temperature (T_o) is measured outside the house. Global radiation on the south façade (ϕ_s) and gross electricity demand (Q_{elec}) are used to estimate effective solar heat gains and internal heat gains, respectively [108]. Heating power provided by the geothermal heat pump (Q_h) is calculated based on the measured air flow rate and temperature difference between supply air and return air. Influence of the ground temperature is neglected since there is a 51 mm (RSI 1.8) insulation between the ground and the slab and no significant temperature difference between the ground and the basement (the measured ground temperature is measured at around 8–11°C). The desired model output is indoor air temperature (T_i) which is an average of room temperatures weighted by the rooms' floor areas.

This house was monitored for two months (January and February in 2011). All measurements are sampled every 0.5 h. The measurements are then divided into one training dataset and one testing dataset (see Fig. A.1). The training dataset includes 30-day data points from January 2011 (number of samples $N = 1440$) while the testing dataset consists of 29-day data points from February 2011.

4.2. Training and testing criteria

Three data-driven models (i.e., RC, ARX, and ANN models) are developed. The prediction error method (PEM) (Subsection 3.2.1) is adopted for model training, where the loss function is defined based on one-step ahead prediction errors:

$$V(\theta) = \frac{1}{N}\sum_{t=1}^{N}\left[e_{t|t-1}(\theta)\right]^2 \tag{10}$$

Minimizing $V(\theta)$ gives parameter estimate $\hat{\theta}$. For RC and AI models, the minimization of $V(\theta)$ is realized by the Levenberg–Marquardt algorithm. For ARX models, the parameters are estimated by linear least squares.

The obtained models are often adopted for one-day ahead prediction, especially in applications like model predictive control. To examine the models' prediction ability, each model is used to predict the indoor air temperature 48 steps into the future (i.e., 24-hour ahead). Then, the predicted temperatures are compared with the measured temperatures using the following criteria:

$$Fit_j = \left(1 - \frac{\sqrt{\sum_{t=1}^{48}\left(y_{j,t} - \hat{y}_{j,t}\right)^2}}{\sqrt{\sum_{t=1}^{48}\left(y_{j,t} - \bar{y}_j\right)^2}}\right)\cdot 100\% \tag{11}$$

where,

Fit_j is the goodness of fit for the j^{th} day;
$y_{j,t}$ is the measured output at time step t in j^{th} day;
$\hat{y}_{j,t}$ is the predicted output at time step t in j^{th} day; and
$\bar{y}_j$ is the average of $y_{j,t}$, i.e., $\bar{y}_j = \frac{1}{48}\sum_{t=1}^{48} y_{j,t}$.

The examination by Eq. (11) is performed on the testing data, which yields *Fit*'s for 29 days. The goodness of fit in percentage informs how closer the data are to the fitted curve compared to a straight line (i.e., $\bar{y}_j$). The larger the *Fit*, the more accurately a model can fit the measurements. Here, instead of displaying all 29 values of *Fit*, its average and standard deviation are used:

$$\overline{Fit} = \frac{1}{29}\sum_{j=1}^{29} Fit_j \tag{12}$$

and

$$Fit\_sd = \sqrt{\frac{1}{28}\sum_{j=1}^{29}\left(Fit_j - \overline{Fit}\right)} \tag{13}$$

A good model should give a high $\overline{Fit}$ and a low *Fit_sd* as much as possible. These two criteria will be used for the following evaluation and selection of suitable models.

4.3. Model development

A list of *RC models* is given in Fig. 4. The potential model elements (for selection) include four inputs (T_o, ϕ_s, Q_h, and Q_{elec}) and three states (T_i, T_m, and T_e). Take Fig. 4(e) for example, T_m and T_e

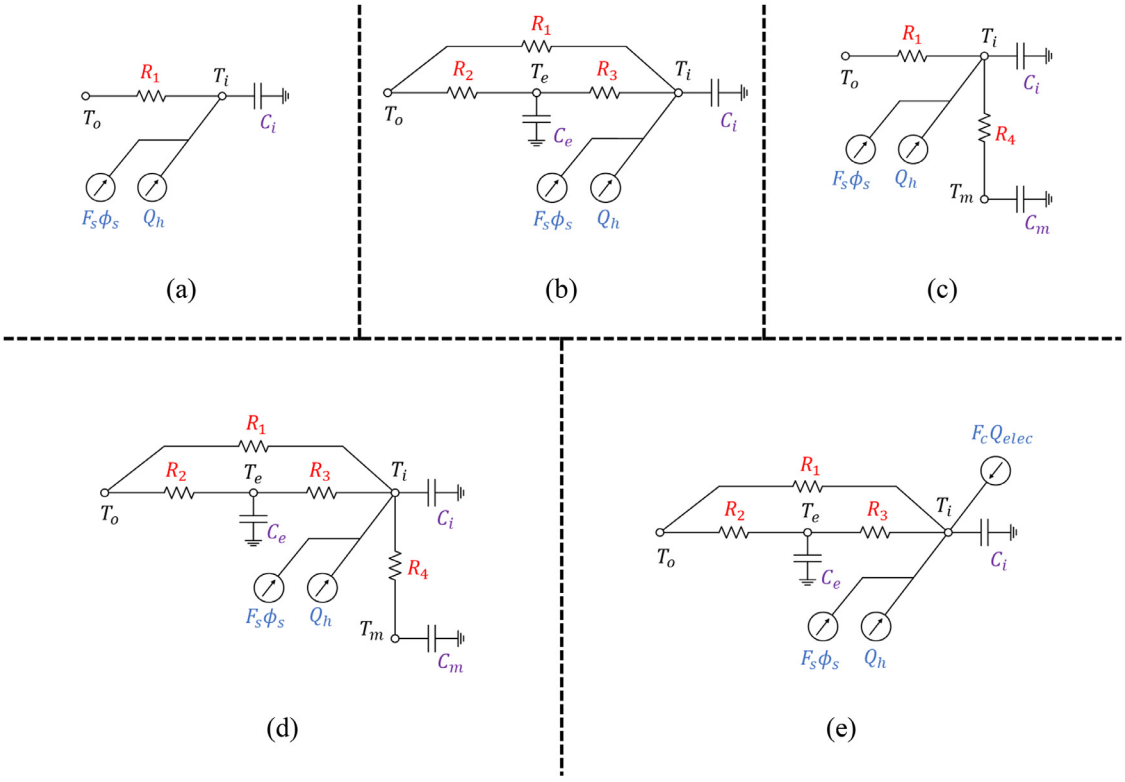


Fig. 4. RC models: (a) $T_o\phi_sQ_h$-T_i; (b) $T_o\phi_sQ_h$-T_iT_e; (c) $T_o\phi_sQ_h$-T_iT_m; (d) $T_o\phi_sQ_h$-$T_iT_mT_e$; (e) $T_o\phi_sQ_hQ_{elec}$-T_iT_e.

represent temperatures of internal concrete mass and building envelop, respectively. C_m and C_e represent their thermal capacitances. C_i denotes thermal capacitance of the indoor air, furniture, etc. The outdoor air temperature (T_o) affects the indoor air temperature (T_i) through a fast response path (R_1) and a slow response path (R_2-C_e-R_3). The fast response path captures the thermal impact through windows and ventilation while the slow response path is mainly to characterize the transient conduction through thick walls, roof, and ceilings. F_s and F_c, are the solar gain factor (i.e., solar aperture) and the internal gain factor that respectively evaluates effective solar gains (from ϕ_s) and effective internal gains (from Q_{elec}) to the indoor. The model structure of Fig. 4(b) is the same as the one in Fig. 2.

To facilitate explanation, an RC model is labelled in the format of RC(inputs-states). The most suitable model is selected by the forward selection technique starting from Model RC($T_o\phi_sQ_h$-T_i). T_o, ϕ_s, and Q_h are considered as initial inputs and Q_{elec} is tested as a model extension. The selection process is shown in Table 5 where the selected model is highlighted. Model RC($T_o\phi_sQ_h$-$T_iT_mT_e$) is not selected though it has slightly better $\overline{Fit}$ and Fit_sd than Model RC($T_o\phi_sQ_h$-T_iT_e). This is because the focus of RC models is to obtain physically interpretable parameters. For Model RC($T_o\phi_sQ_h$-$T_iT_mT_e$), some parameters are estimated to have unreasonable values and considerable uncertainties, hence are not interpretable.

Table 5 indicates that extending the RC model to include Q_{elec} makes a negligible improvement to either its training or testing performance. This is because the internal heat gains are insignificant compared to Q_h, particularly in the wintertime like January and February. Thus, Q_{elec} is no longer considered for the following development of ARX and ANN models.

Table 5
Selection of a suitable RC model.

RC models			Training	Testing	
Inputs	States	Cases in Fig. 4	V	$\overline{Fit}$%	Fit_sd%	
$T_o\phi_sQ_h$	T_i	(a)	0.3186	−29.64	59.62
	T_iT_e	(b)	0.0098	72.60	12.12
	T_iT_m	(c)	0.0098	68.90	13.24
	$T_iT_mT_e$	(d)	0.0099	73.57	10.82
$T_o\phi_sQ_hQ_{elec}$	T_iT_e	(e)	0.0100	72.18	12.32

The development of an *ARX model* for the same thermal zone is now illustrated. T_o, ϕ_s, and Q_h are chosen as inputs in this illustration. The ARX model is labelled in the format of ARX(na, [nb_1 nb_2 nb_3]), where na and [nb_1 nb_2 nb_3] are output and input delays, respectively. Explicitly, the model can be expressed as:

$$\left(1+a_1z^{-1}+\ldots+a_{na}z^{-na}\right)T_{i,t}=\begin{bmatrix}b_{1,0}+b_{1,1}z^{-1}+\ldots+b_{1,nb_1}z^{-nb_1}\\ b_{2,0}+b_{2,1}z^{-1}+\ldots+b_{2,nb_2}z^{-nb_2}\\ b_{3,0}+b_{3,1}z^{-1}+\ldots+b_{3,nb_3}z^{-nb_3}\end{bmatrix}\begin{bmatrix}T_{o,t} & \phi_{s,t} & Q_{h,t}\end{bmatrix}+e_t$$

The backward selection routine (Subsection 3.3.1) is employed for selecting a suitable model. The selected Model ARX(10, [1 10 10]), as highlighted in Table 6, has $\overline{Fit}$ and Fit_sd less than 1% different from the model with the largest structure. Further reducing its order will cause an apparent decrease in the testing performance.

Finally, the development of an *ANN model* for the same thermal zone is now illustrated. Again, T_o, ϕ_s, and Q_h are chosen as inputs in this illustration. A feedforward ANN model (see Table 3 for the model structure) is labelled in the format of ANN (nl, 0: ud, 1: yd). All input delays are set the same ($ud_1 = ud_2 = ud_3 = ud$) which

Table 6
Selection of a suitable ARX model.

ARX models			Training	Testing	
Inputs	*na*	$[nb_1\ nb_2\ nb_3]$	*V*	$\overline{Fit}$%	*Fit_sd%*
$T_o\phi_sQ_h$	12	[12 12 12]	0.0044	76.13	12.42
		[1 10 10]	0.0045	76.25	13.03
		[1 10 8]	0.0046	75.52	14.08
	10	[1 10 10]	0.0045	76.25	13.02
	8	[1 10 10]	0.0046	75.64	14.15

Table 7
Selection of a suitable ANN model.

ANN models				Training	Testing	
Inputs	*nl*	0: *ud*	1: *yd*	*V*	$\overline{Fit}$%	*Fit_sd%*
$T_o\phi_sQ_h$	4	0:2	1:2	0.0024	77.85	11.38
		0:3	1:2	0.0016	77.56	10.98
		0:2	1:3	0.0023	78.36	7.00
		0:2	1:4	0.0017	77.71	10.56
	5	0:2	1:3	0.0016	77.02	10.80

Table 8
Summary of the testing performance of selected models.

Selected models	$\overline{Fit}$%	*Fit_sd%*
RC ($T_o\phi_sQ_h$ - T_iT_e)	72.60	12.12
ARX (10, [1 10 10])	76.25	13.02
ANN (4, 0:2, 1:3)	78.36	7.00

leaves the training algorithm to adjust the relative importance (i.e., weight) of each delay. Hyperbolic tangent function (tanh) is adopted as the activation function. Before training, the weights and biases are initialized with random small values. Since a lot of weights and biases are being used, there may exist multiple local minima. To overcome this problem, each model is initialized and trained for 20 times from which the one with the best testing performance is screened out. Moreover, to avoid overfitting, the early stopping [69] is employed and 20% of the training data is used for cross-validation. Based on the above settings, model selection is performed forwardly starting from ANN (4, 0:2, 1:2). The selected model is ANN (4, 0:2, 1:3) which gives the highest $\overline{Fit}$ and the lowest *Fit_sd* (see Table 7).

4.4. Analysis of results

Autocorrelations and cumulative periodograms of the training residuals ($e = y - \hat{y}$) are plotted for the selected RC, ARX and ANN models in Fig. 5. For each plot, the 95% confidence interval under the null hypothesis that the residuals are white noise is also shown (by parallel lines). There is an explicitly low dependency of autocorrelations on lags and of periodograms on frequencies for all three models, with RC($T_o\phi_sQ_h$-T_iT_e) exhibits autocorrelations and cumulative periodogram outside the confidence region slightly more than ARX(10, [1,10]) and ANN(4, 0:2, 1:3), It is reasonable to accept that the residuals are white noise, suggesting that the thermal dynamics contained in the training dataset are well modeled [29,41].

The testing performances of the selected models are summarized in Table 8. The ANN model, with the highest $\overline{Fit}$% and the lowest *Fit_sd%*, demonstrates strong ability and reliability in one-day ahead prediction. This is because the ANN model accounts for nonlinear input-output relations and interactions between different inputs. Comparatively, the ARX model also exhibits excellent prediction ability but its performance comes at the cost of large model order (input/output delays as large as 10) and large variation (*Fit_sd%* is the highest). Its prediction performance is not as consistent as the ANN model. A thorough comparison of ARX and ANN models can also be found in [60], which derives a rather similar conclusion that a nonlinear ANN model is more suitable than a linear ARX model for the prediction of room temperatures. As for the RC model in this case, it has the smallest but acceptable *Fit*. If the modeling objective is more on obtaining a physically meaningful model for characterizing building thermal dynamics, the RC model is the best choice. A better *Fit* value may be obtained by extending its model structure; however, the parameter estimates may become physically unreasonable.

The predicted indoor air temperatures by the selected models are plotted against the measurements for four representative days in the testing period (Fig. 6). These four days are chosen to represent four different cases: (a) all models have testing performances around average $\overline{Fit}$%; (b) the ANN model notably outperforms the RC and ARX models; (c) all models have testing performances above average; (d) all models have testing performances below average. These four days cover all possible situations that occurred in the testing period.

For most days, the ANN model has higher prediction accuracy and can capture the dynamics overlooked by the RC and ARX models (e.g., cases in Fig. 6 (a,b)). On days like in Fig. 6 (b,d), though the prediction accuracy is less favorable, the prediction errors are within 0.5 °C at most of the time. Besides, most reported model performance in the literature varies around *Fit*-70% for RC and ARX models [25,45,59]. It can be concluded that the prediction ability of the selected models is fairly acceptable. If further higher accuracy is required, it may be necessary to design independently a 48-step predictor [79]. In this case study, we assumed one-step prediction horizon in training but tested the model for 48-step (24 h) ahead prediction. Although it gives a satisfactory result, instead, using a 48-step ahead horizon for the objective function can focus the model on one-day ahead prediction, hence improves the prediction accuracy.

5. Discussion

RC, TF, and AI models are manifestly different in their theories and numerical formulations. The RC models are constructed based on a series of ordinary differential equations that characterize thermal dynamics in buildings. The appropriate design of RC models requires an in-depth understanding of the thermal systems of interest. Moreover, since the ordinary differential equations are set up in continuous-time format, but measured input and output data are at discrete times, equation discretization is required. In contrast, the TF models are more straightforward to develop. They simply use rational functions or polynomials to incorporate the input and output variables. Since the rational functions or polynomials are already in discrete-time, discretization is not needed. Compared to the linear RC and TF models, the AI models can be regarded as nonlinear regression models built upon machine learning techniques. With intricate inner structures, the AI models can implicitly account for nonlinear input-output relationships and complex interactions between inputs.

Another difference between the three types of models arises from their physical interpretability. Both RC and TF models have their counterparts in the forward approach, e.g., thermal networks and comprehensive room transfer functions. For RC models, the equivalent thermal parameters (e.g., thermal resistances and capacitances) have intuitive physical meanings. Hence they are suitable for explanatory purposes on the thermal dynamics. However, the physical interpretation of TF models is not straightforward. Only some stationary values can be retrieved, such as the thermal transmittance and solar heat gain coefficient. For AI models, they are impossible for any physical interpretation. This is because the AI

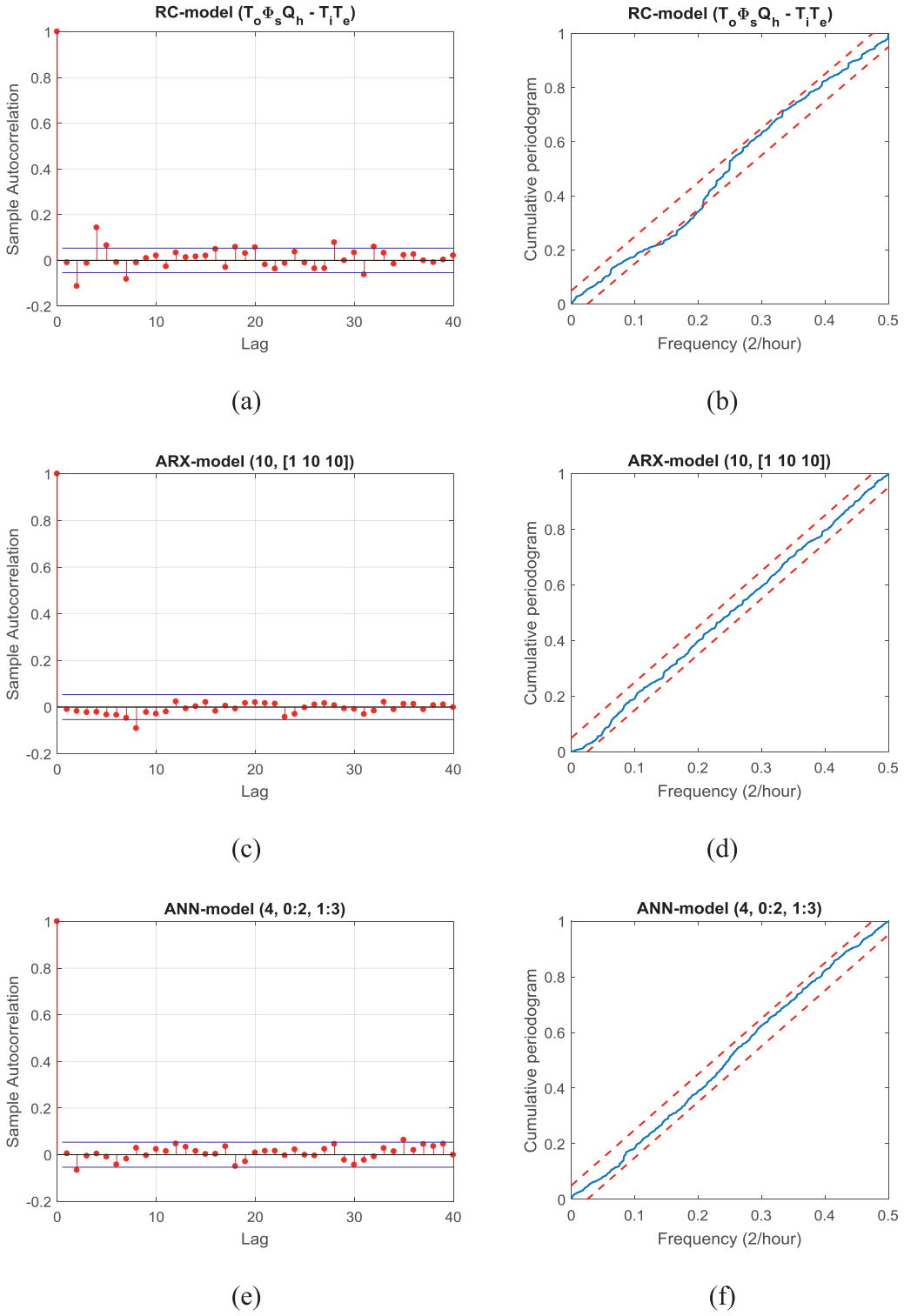


Fig. 5. Autocorrelations and cumulative periodograms of residuals for the selected models (plotted with 95% confidence intervals).

models are not built based on physical principles and their structures are too mathematically complicated to be physically understood.

In terms of model training, TF models (ARX and SS models in particular) are computationally more efficient than RC models. This is because in RC models, the ordinary differential equations and their discretization can make the outputs highly nonlinear with respect to the unknown parameters. Due to the same reason, more local minimums may occur in RC models' training. Comparatively, the ARX or SS model allows for linear least squares or subspace identification for model training so that the loss function is guaranteed to be globally optimized. As for AI models, other concerns like overfitting can be more crucial than computational efficiency.

Most ANN models are trained using backpropagation-based searching algorithms. Since there are often a lot of parameters (e.g., weights and biases) to be estimated, overfitting can become

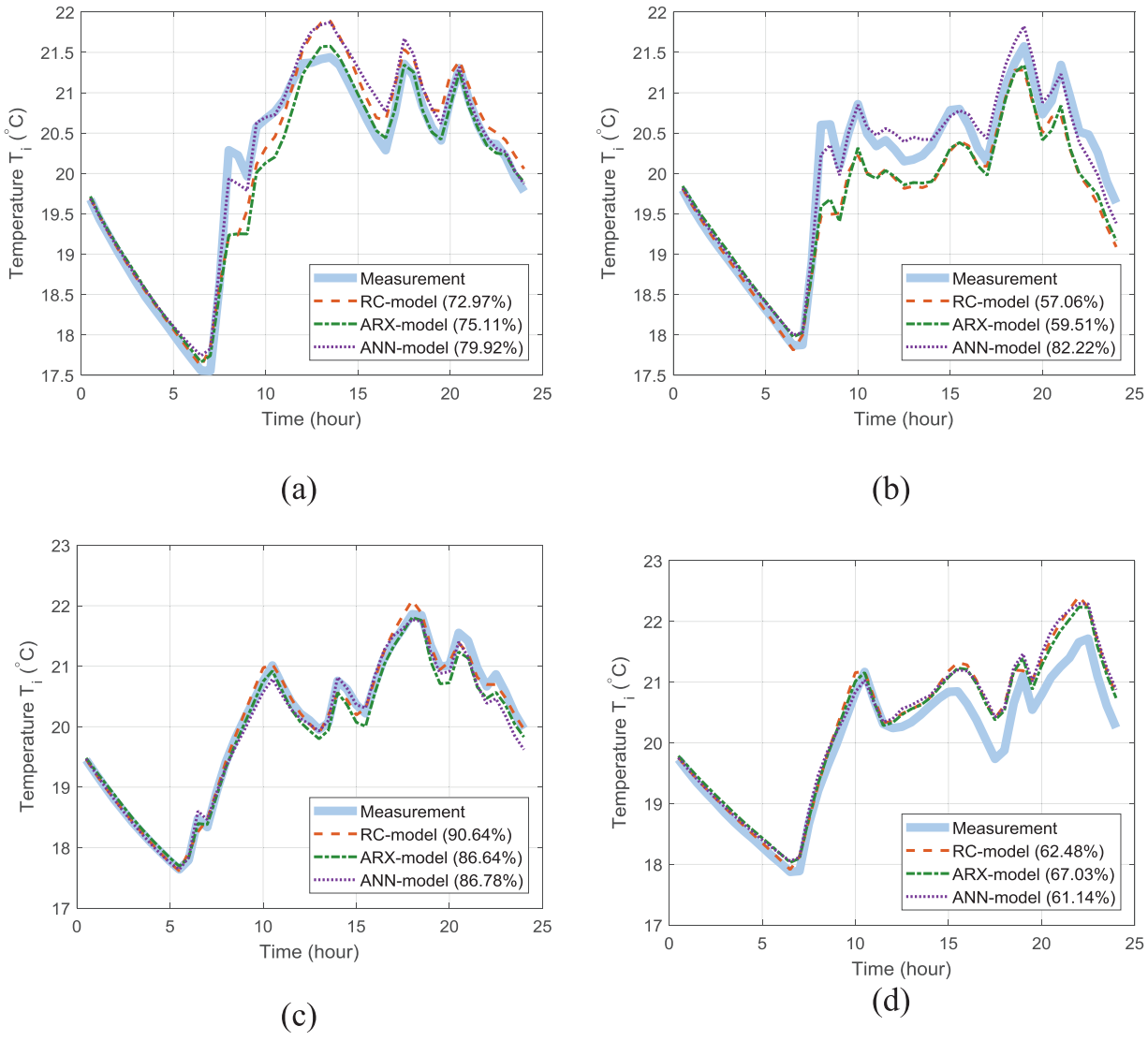


Fig. 6. Testing results on four representative days (Feb 25, Feb 27, Feb 10, and Feb 11 in sequence): in parenthesis is the *Fit* of each model for that day.

a serious problem that influences the model's reproductivity on new datasets. A conventional technique to avoid overfitting is early stopping which uses a portion of the training dataset for cross-validation and ends training when the cross-validation error starts to increase [69]. The overfitting problem can also be overcome by including the model complexity (e.g., number of neurons) in the loss function to minimize [87]. However, not all AI models are subjected to overfitting. For example, the GRNN models can be trained in one pass through the data with no need for any iterative algorithm [91,92]. An essential advantage of the GRNN models is fast learning. SVM models are trained with the structural risk minimization principle which defines a trade-off between the model complexity and quality of fitting the training data [98]. This intrinsic feature of SVMs also prevents the model from being overfitted.

6. Conclusion

Data-driven modeling of building thermal dynamics consists of three phases: modeling, training, and selecting. TF, RC and AI models are three main categories of data-driven models. RC models are most suitable for physical interpretation, TF models are the easiest to formulate, and the AI models can conveniently manage nonlinearity and complex interactions. In the training phase, the prediction error method (PEM) and maximum likelihood estimation (MLE) are two popular methods to build up loss functions. Unknown parameters in the data-driven models are estimated by minimizing the values of the loss functions, which is numerically accomplished by local or global search algorithms. In the selecting phase, the most suitable model structure is selected through a forward or backward selecting procedure. Model selection balances a model's prediction accuracy against its complexity. Statistical quality criteria and residual analysis are used to validate, test, and compare model candidates with different inputs and structures.

Data-driven modeling is illustrated by the case study of a single-zone house. Three models, i.e., an RC model, an ARX model, and an ANN model, are developed to predict the indoor air temperature of the house. After training and selecting, all models exhibit favourable prediction ability, with the ANN model generally outperforming the RC and ARX models. On the other hand, both the ARX and RC models can be used to derive important thermal properties of the house, but the RC model serves better for explanatory purposes. Finally, the ARX model has the advantage of being trained by linear least squares that guarantees a global optimal to be found.

Declaration of Competing Interest

None.

Acknowledgement

The authors would like to acknowledge Dr. Andreas K. Athienitis and his research team for providing the data. The research work presented in the paper is financially supported by the University of Alberta.

Appendix A. Datasets

Fig. A.1.

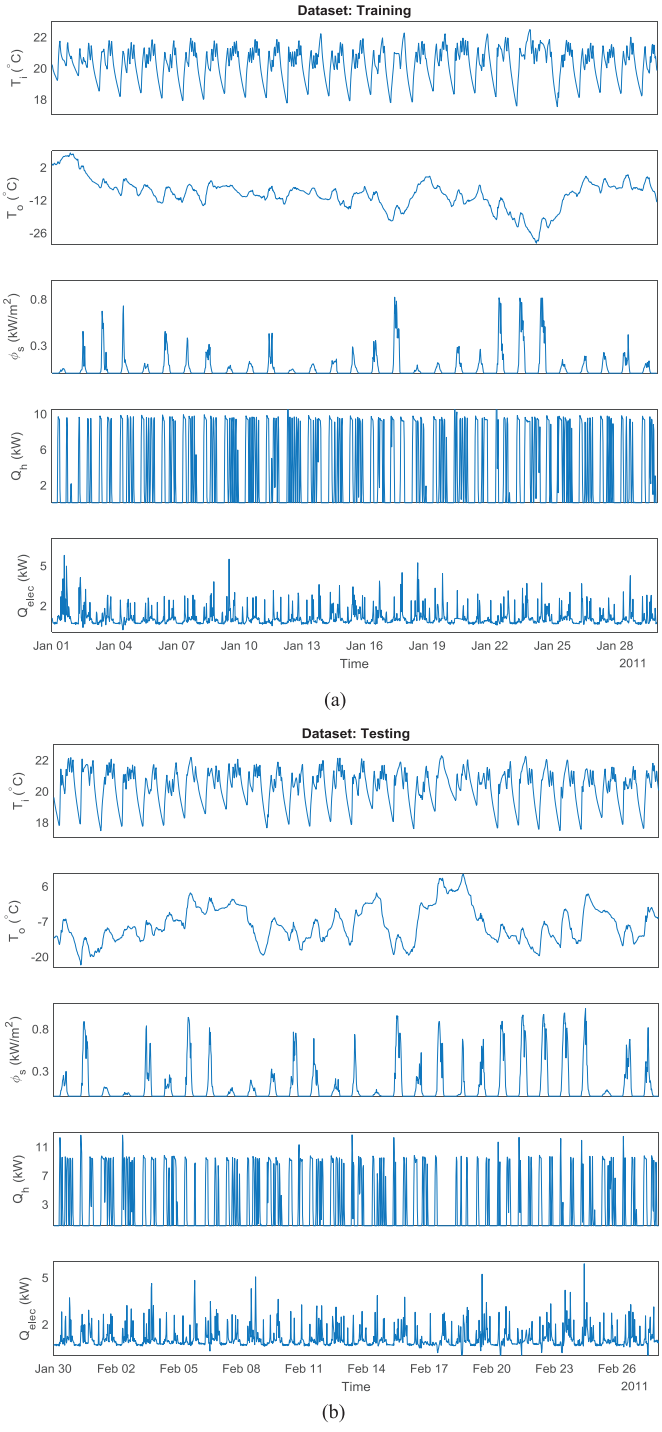


Fig. A.1. Measured data for (a) training, and (b) testing. Sampling interval: 0.5 h.

References

[1] International Energy Agency, Transition to Sustainable buildings: Strategies and Opportunities to 2050, International Energy Agency, Paris, 2013.
[2] L. Pérez-Lombard, J. Ortiz, C. Pout, A review on buildings energy consumption information, Energy Build. 40 (2008) 394–398.
[3] T.A. Reddy, Applied Data Analysis and Modeling for Energy Engineers and Scientists, Springer, New York, 2011.
[4] ASHRAE, 2017 ASHRAE Handbook Fundamentals (2017).
[5] Y. Ma, Model Predictive Control for Energy Efficient Buildings (Doctoral Thesis), UC Berkeley, 2012.
[6] K. Lee, J.E. Braun, Model-based demand-limiting control of building thermal mass, Build. Environ. 43 (2008) 1633–1646.
[7] M. Sourbron, C. Verhelst, L. Helsen, Building models for model predictive control of office buildings with concrete core activation, J. Build. Perform. Simul. 6 (2013) 175–198.
[8] W.Y. Lee, et al., Fault detection in an air-handling unit using residual and recursive parameter identification methods, Winter meeting of American Society of Heating, Refrigeration and Air Conditioning Engineers, Atlanta, GA (United States), 17-21 Feb 1996, 1996.
[9] J. Wang, H. Wang, Y. Chen, C.W.H. Chan, J. Qin, Online model-based fault detection and diagnosis strategy for VAV air handling units, Energy Build. 55 (2012) 252–263.
[10] M. Yalcintas, Energy-savings predictions for building-equipment retrofits, Energy Build. 40 (2008) 2111–2120.
[11] H.A. Nielsen, H. Madsen, Modeling the heat consumption in district heating systems using a grey-box approach, Energy Build. 38 (2006) 63–71.
[12] W.J. Cole, K.M. Powell, E.T. Hale, T.F. Edgar, Reduced-order residential home modeling for model predictive control, Energy Build. 74 (2014) 69–77.
[13] I. Hazyuk, C. Ghiaus, D. Penhouet, Optimal temperature control of intermittently heated buildings using model predictive control: part I – Building modeling, Build. Environ. 51 (2012) 379–387.
[14] I. Hazyuk, C. Ghiaus, D. Penhouet, Optimal temperature control of intermittently heated buildings using model predictive control: part II – Control algorithm, Build. Environ. 51 (2012) 388–394.
[15] O. Mejri, E. Palomo Del Barrio, N. Ghrab-Morcos, Energy performance assessment of occupied buildings using model identification techniques, Energy Build. 43 (2011) 285–299.
[16] C. Ghiaus, A. Chicinas, C. Inard, Grey-box identification of air-handling unit elements, Control Eng. Pract. 15 (2007) 421–433.
[17] S. Prívara, Z. Váňa, E. Žáčeková, J. Cigler, Building modeling: selection of the most appropriate model for predictive control, Energy Build. 55 (2012) 341–350.
[18] J. Hu, P. Karava, A state-space modeling approach and subspace identification method for predictive control of multi-zone buildings with mixed-mode cooling, in: Proceedings of International High Performance Buildings Conference, 2014.
[19] R. Sonderegger, Diagnostic tests determining the thermal response of a house, ASHRAE meeting, Atlanta, GA, USA, 29 Jan 1978, 1977.
[20] H. Madsen, J. Holst, Estimation of continuous-time models for the heat dynamics of a building, Energy Build. 22 (1995) 67–79.
[21] A. Saberi Derakhtenjani, J.A. Candanedo, Y. Chen, V.R. Dehkordi, A.K. Athienitis, Modeling approaches for the characterization of building thermal dynamics and model-based control: a case study, Sci. Tech. Built Environ. 21 (2015) 824–836.
[22] Z. Liao, A.L. Dexter, A simplified physical model for estimating the average air temperature in multi-zone heating systems, Build. Environ. 39 (2004) 1013–1022.
[23] R. Bălan, J. Cooper, K. Chao, S. Stan, R. Donca, Parameter identification and model based predictive control of temperature inside a house, Energy Build. 43 (2011) 748–758.
[24] J.E. Braun, N. Chaturvedi, An inverse gray-box model for transient building load prediction, HVAC&R Res. 8 (2002) 73–99.
[25] Q. Zhou, S. Wang, X. Xu, F. Xiao, A grey-box model of next-day building thermal load prediction for energy-efficient control, Int. J. Energy Res. 32 (2008) 1418–1431.
[26] S. Wang, X. Xu, Simplified building model for transient thermal performance estimation using GA-based parameter identification, Int. J. Therm. Sci. 45 (2006) 419–432.
[27] J. Date, et al., Control-oriented modeling of thermal zones in a house: a multi-level approach, International High Performance Buildings Conference, 2016.
[28] P. Biddulph, V. Gori, C.A. Elwell, C. Scott, C. Rye, R. Lowe, T. Oreszczyn, Inferring the thermal resistance and effective thermal mass of a wall using frequent temperature and heat flux measurements, Energy Build. 78 (2014) 10–16.
[29] P. Andersen, M. Jiménez, H. Madsen, C. Rode, Characterization of heat dynamics of an arctic low-energy house with floor heating, Build. Simul. 7 (2014) 595–614.
[30] P.H. Baker, H.A.L. van Dijk, PASLINK and dynamic outdoor testing of building components, Build. Environ. 43 (2008) 143–151.
[31] M.J. Jiménez, B. Porcar, M.R. Heras, Estimation of building component UA and gA from outdoor tests in warm and moderate weather conditions, Sol. Energy 82 (2008) 573–587.
[32] J.M. Penman, Second order system identification in the thermal response of a working school, Build. Environ. 25 (1990) 105–110.

[33] D.A. Coley, J.M. Penman, Second order system identification in the thermal response of real buildings. Paper II: recursive formulation for on-line building energy management and control, Build. Environ. 27 (1992) 269–277.
[34] J. Date, A.K. Athienitis, M. Fournier, A study of temperature set point strategies for peak power reduction in residential buildings, Energy Procedia 78 (2015) 2130–2135.
[35] S.F. Fux, A. Ashouri, M.J. Benz, L. Guzzella, EKF based self-adaptive thermal model for a passive house, Energy Build. 68 (2014) 811–817.
[36] R.E. Hedegaard, S. Petersen, Evaluation of grey-box model parameter estimates intended for thermal characterization of buildings, Energy Procedia 132 (2017) 982–987.
[37] J.E. Seem, S.A. Klein, W.A. Beckman, J.W. Mitchell, Transfer functions for efficient calculation of multidimensional transient heat transfer, J. Heat Transfer. 111 (1989) 5.
[38] J. Široký, F. Oldewurtel, J. Cigler, S. Prívara, Experimental analysis of model predictive control for an energy efficient building heating system, Appl. Energy 88 (2011) 3079–3087.
[39] G.F. Franklin, J.D. Powell, M.L. Workman, Digital Control of Dynamic Systems, Addison Wesley, Menlo Park, Calif., 1998.
[40] H. Harb, N. Boyanov, L. Hernandez, R. Streblow, D. Müller, Development and validation of grey-box models for forecasting the thermal response of occupied buildings, Energy Build. 117 (2016) 199–207.
[41] P. Bacher, H. Madsen, Identifying suitable models for the heat dynamics of buildings, Energy Build. 43 (2011) 1511–1522.
[42] H. Kwakernaak, R. Sivan, Linear Optimal Control Systems, Wiley, New York, 1972 <>.
[43] S. Prívara, J. Široký, L. Ferkl, J. Cigler, Model predictive control of a building heating system: the first experience, Energy Build. 43 (2011) 564–572.
[44] K.J. Keesman, System Identification: An Introduction, Springer, London; New York, 2011.
[45] G. Mustafaraj, J. Chen, G. Lowry, Development of room temperature and relative humidity linear parametric models for an open office using bms data, Energy Build 42 (, 3) (2010) 348–356.
[46] M.J. Jiménez, H. Madsen, K.K. Andersen, Identification of the main thermal characteristics of building components using MATLAB, Build. Environ. 43 (2008) 170–180.
[47] W.J.N. Turner, A. Staino, B. Basu, Residential HVAC fault detection using a system identification approach, Energy Build. 151 (2017) 1–17.
[48] P. Moroşan, R. Bourdais, D. Dumur, J. Buisson, Building temperature regulation using a distributed model predictive control, Energy Build. 42 (2010) 1445–1452.
[49] P.R. Armstrong, S.B. Leeb, L.K. Norford, Control with building mass – Part I: thermal response model, ASHRAE Trans. 112 (2006) 449–461.
[50] P.R. Armstrong, S.B. Leeb, L.K. Norford, Control with building mass-part II: simulation, ASHRAE Trans. 112 (2006) 462.
[51] P.R. Armstrong, Model identification with application to building control and fault detection (2004).
[52] T.Y. Chen, Application of adaptive predictive control to a floor heating system with a large thermal lag, Energy Build. 34 (2002) 45–51.
[53] T.Y. Chen, Real-time predictive supervisory operation of building thermal systems with thermal mass, Energy Build. 33 (2001) 141–150.
[54] R.Z. Freire, G.H.C. Oliveira, N. Mendes, Predictive controllers for thermal comfort optimization and energy savings, Energy Build. 40 (2008) 1353–1365.
[55] J.E. Seem, Modeling of heat transfer in buildings (1987).
[56] T. McKelvey, Identification of State-Space Models from Time and Frequency Data, Univ Linköping, Linköping, 1995.
[57] J.A. Candanedo, V.R. Dehkordi, M. Stylianou, Model-based predictive control of an ice storage device in a building cooling system, Appl. Energy 111 (2013) 1032.
[58] S. Royer, et al., Black-box modeling of buildings thermal behavior using system identification, in: Proceedings of the 19th IFAC World Congress, 2014.
[59] G.J. Ríos-Moreno, M. Trejo-Perea, R. Castañeda-Miranda, V.M. Hernández-Guzmán, G. Herrera-Ruiz, Modeling temperature in intelligent buildings by means of autoregressive models, Autom. Constr. 16 (2007) 713–722.
[60] G. Mustafaraj, G. Lowry, J. Chen, Prediction of room temperature and relative humidity by autoregressive linear and nonlinear neural network models for an open office, Energy Build. 43 (2011) 1452–1460.
[61] K. Yun, R. Luck, P.J. Mago, H. Cho, Building hourly thermal load prediction using an indexed ARX model, Energy Build. 54 (2012) 225.
[62] H.A. Nielsen, H. Madsen, Predicting the heat consumption in district heating systems using meteorological forecasts (2000).
[63] P. Bacher, H. Madsen, H.A. Nielsen, B. Perers, Short-term heat load forecasting for single family houses, Energy Build. 65 (2013) 101.
[64] A. Rabl, Parameter estimation in buildings: methods for dynamic analysis of measured energy use, J. Sol. Energy Eng. 110 (1988) 52.
[65] K. Subbarao, Building parameters and their estimation from performance monitoring (1985).
[66] T.Y. Chen, A.K. Athienitis, Investigation of practical issues in building thermal parameter estimation, Build. Environ. 38 (2003) 1027–1038.
[67] John McCarthy, What is artificial intelligence, Computer Science Department, Stanford University, 2007.
[68] D.J.C. MacKay, Bayesian non-linear modeling for the prediction competition, in: G.R. Heidbreder (Ed.), Maximum Entropy and Bayesian Methods: Santa Barbara, Springer, California, U.S.A.Netherlands, 1993, pp. 221–234. Dordrecht, 1996.
[69] J.F. Kreider, D.E. Claridge, P. Curtiss, R. Dodier, J.S. Haberl, M. Krarti, Building energy use prediction and system identification using recurrent neural networks, J. Sol. Energy Eng. 117 (1995) 161.
[70] M. Kawashima, et al., Hourly thermal load prediction for the next 24 h by ARIMA, EWMA, LR and an artificial neural network, American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE) winter meeting and exhibition, Chicago, IL (United States), 28 Jan - 1 Feb 1995, 1995.
[71] J.F. Kreider, Predicting hourly building energy use: the great energy predictor shootout – Overview and discussion of results, 1994 American Society of Heating, Refrigerating, and Air Conditioning Engineers (ASHRAE) annual meeting, Orlando, FL (United States), 25-29 Jun 1994, 1994.
[72] J.S. Haberl, S. Thamilseran, Great energy predictor shootout II: measuring retrofit savings – overview and discussion of results, 1996 annual meeting of the American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE), Inc., San Antonio, TX (United States), 22-26 Jun 1996, 1996.
[73] M.Q. Raza, A. Khosravi, A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings, Renew. Sustain. Energy Rev. 50 (2015) 1352–1372.
[74] C. Deb, et al., A review on time series forecasting techniques for building energy consumption, Renew. Sustain. Energy Rev. 74 (2017) 902–924.
[75] M.A. Mat Daut, M.Y. Hassan, H. Abdullah, H.A. Rahman, M.P. Abdullah, F. Hussin, Building electrical energy consumption forecasting analysis using conventional and artificial intelligence methods: a review, Renew. Sustain. Energy Rev. (2016).
[76] G. Dudek, Artificial immune system for short-term electric load forecasting, in: Artificial Intelligence and Soft Computing – ICAISC 2008, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 1007–1017.
[77] G. Dudek, Short-term load forecasting using random forests (2015) 821–828.
[78] Marcel van Gerven, Sander Bohte, Artificial Neural Networks as Models of Neural Information Processing, Frontiers Media SA, 2018.
[79] S. Karatasou, M. Santamouris, V. Geros, Modeling and predicting building's energy use with artificial neural networks: methods and results, Energy Build. 38 (2006) 949–958.
[80] R. Yokoyama, T. Wakui, R. Satake, Prediction of energy demands using neural network with model identification by global optimization, Energy Convers. Manage. 50 (2009) 319–327.
[81] P.A. González, J.M. Zamarreño, Prediction of hourly energy consumption in buildings based on a feedback artificial neural network, Energy Build. 37 (2005) 595–601.
[82] P.A. González Lanza, J.M. Zamarreño Cosme, A short-term temperature forecaster based on a state space neural network, Eng. Appl. Artif. Intell. 15 (2002) 459–464.
[83] S.A. Kalogirou, M. Bojic, Artificial neural networks for the prediction of the energy consumption of a passive solar building, Energy 25 (2000) 479–491.
[84] G. Zhang, B. Eddy Patuwo, M.Y. Hu, Forecasting with artificial neural networks::: the state of the art, Int. J. Forecast. 14 (1998) 35–62.
[85] J. Yang, H. Rivard, R. Zmeureanu, On-line building energy prediction using adaptive artificial neural networks, Energy Build. 37 (2005) 1250–1259.
[86] R.H. Dodier, G.P. Henze, Statistical analysis of neural networks as applied to building energy prediction, J. Sol. Energy Eng. 126 (2004) 592.
[87] A.E. Ruano, E.M. Crispim, E.Z.E. Conceição, M.M.J.R. Lúcio, Prediction of building's temperature using neural networks models, Energy Build. 38 (2006) 682–694.
[88] P.M. Ferreira, A.E. Ruano, S. Silva, E.Z.E. Conceição, Neural networks based predictive control for thermal comfort and energy savings in public buildings, Energy Build. 55 (2012) 238–251.
[89] Weijie Mai, et al., Electric load forecasting for large office building based on radial basis function neural network, PESGM (2014) 1–5.
[90] Q. Li, Q. Meng, J. Cai, H. Yoshino, A. Mochida, Predicting hourly cooling load in the building: a comparison of support vector machine and different artificial neural networks, Energy Convers. Manage. 50 (2009) 90–96.
[91] D.F. Specht, A general regression neural network, IEEE Trans. Neural Netw. 2 (1991) 568–576.
[92] W. Lee, J.M. House, N. Kyong, Subsystem level fault diagnosis of a building's air-handling unit using general regression neural networks, Appl. Energy 77 (2004) 153–170.
[93] J.W. Moon, S.K. Jung, Y. Kim, S. Han, Comparative study of artificial intelligence-based building thermal control methods – Application of fuzzy, adaptive neuro-fuzzy inference system, and artificial neural network, Appl. Therm. Eng. 31 (2011) 2422–2429.
[94] Z. Du, B. Fan, J. Chi, X. Jin, Sensor fault detection and its efficiency analysis in air handling unit using the combined neural networks, Energy Build. 72 (2014) 157–166.
[95] M. Mohandes, Support vector machines for short-term electrical load forecasting, Int. J. Energy Res. 26 (2002) 335–345.
[96] Q. Li, Q. Meng, J. Cai, H. Yoshino, A. Mochida, Applying support vector machine to predict hourly cooling load in the building, Appl. Energy 86 (2009) 2249–2256.
[97] F. Zhang, C. Deb, S.E. Lee, J. Yang, K.W. Shah, Time series forecasting for building energy consumption using weighted support vector regression with differential evolution optimization technique, Energy Build. 126 (2016) 94–103.
[98] V.N. Vapnik, The Nature of Statistical Learning Theory, 2nd ed., Springer, New York, 2000 [u.a.].
[99] J. Che, J. Wang, Short-term load forecasting using a kernel-based support vector regression combination model, Appl. Energy 132 (2014) 602–609.

[100] B. Dong, C. Cao, S.E. Lee, Applying support vector machines to predict building energy consumption in tropical region, Energy Build. 37 (2005) 545–553.
[101] B. Schölkopf, A.J. Smola, Learning with Kernels, MIT Press, Cambridge, 2002 Mass. [u.a.].
[102] Y. Yao, Z. Lian, S. Liu, Z. Hou, Hourly cooling load prediction by a combined forecasting model based on analytic hierarchy process, Int. J. Therm. Sci. 43 (2004) 1107–1118.
[103] C. Fan, F. Xiao, S. Wang, Development of prediction models for next-day building energy consumption and peak power demand using data mining techniques, Appl. Energy 127 (2014) 1–10.
[104] T. Senjyu, P. Mandal, K. Uezato, T. Funabashi, Next day load curve forecasting using hybrid correction method, TPWRS 20 (2005) 102–109.
[105] D.K. Chaturvedi, A.P. Sinha, O.P. Malik, Short term load forecast using fuzzy logic and wavelet transform integrated generalized neural network, Int. J. Elect. Power Energy Syst. 67 (2015) 230–237.
[106] M. Rana, I. Koprinska, Forecasting electricity load with advanced wavelet neural networks, Neurocomputing 182 (2016) 118–132.
[107] P. Fazenda, P. Lima, P. Carreira, Context-based thermodynamic modeling of buildings spaces, Energy Build. 124 (2016) 164–177.
[108] G. Reynders, J. Diriken, D. Saelens, Quality of grey-box models and identified parameters as function of the accuracy of input and observation signals, Energy Build. 82 (2014) 263–274.
[109] F. Oldewurtel, D. Sturzenegger, M. Morari, Importance of occupancy information for building climate control, Appl. Energy 101 (2013) 521–532.
[110] Yashen Lin, et al., Issues in identification of control-oriented thermal models of zones in multi-zone buildings, CDC (2012) 6932–6937.
[111] A. Deconinck, S. Roels, Is stochastic grey-box modeling suited for physical properties estimation of building components from on-site measurements? J. Building Phys. 40 (2017) 444–471.
[112] L. Ljung, System Identification, Prentice Hall, Upper Saddle River, N.J., 1999.
[113] R.B. Gopaluni, R.S. Patwardhan, S.L. Shah, MPC relevant identification—-tuning the noise model, J. Process Control 14 (2004) 699–714.
[114] R.B. Gopaluni, R.S. Patwardhan, S.L. Shah, Bias distribution in MPC relevant identification, IFAC Proc. Vol. 35 (2002) 435–440.
[115] J. Rehor, V. Havlena, Grey-box model identification – control relevant approach, IFAC Proc. Vol. 43 (2010) 117–122.
[116] J. Zhao, Y. Zhu, R. Patwardhan, Identification of k-step-ahead prediction error model and MPC control, J. Process Control 24 (2014) 48–56.
[117] S. Prívara, J. Cigler, Z. Váňa, F. Oldewurtel, E. Žáčeková, Use of partial least squares within the control relevant identification for buildings, Control Eng. Pract. 21 (2013) 113–121.
[118] T. Söderström, P. Stoica, Some properties of the output error method, Automatica 18 (1982) 93–99.
[119] Y. Tomita, A.A.H. Damen, Paul M.J. Van Den Hof, Equation error versus output error methods, Ergonomics 35 (1992) 551–564.
[120] D. Eckhard, A.S. Bazanella, C.R. Rojas, H. Hjalmarsson, Cost function shaping of the output error criterion, Automatica 76 (2017) 53–60.
[121] N.R. Kristensen, H. Madsen, S.B. Jørgensen, Parameter estimation in stochastic grey-box models, Automatica 40 (2004) 225–237.
[122] K.J. Åström, Maximum likelihood and prediction error methods, Automatica 16 (1980) 551–574.
[123] T. Soderstrom, H. Fan, B. Carlsson, S. Bigi, Least squares parameter estimation of continuous-time ARX models from discrete-time data, TAC 42 (1997) 659–673.
[124] L. Ljung, Some classical and some new ideas for identification of linear systems, J. Control Autom. Electr. Syst. 24 (2013) 3–10.
[125] S.J. Qin, An overview of subspace identification, Comput. Chem. Eng. 30 (2006) 1502–1513.
[126] P. Van Overschee, B. De Moor, Subspace Identification for Linear Systems: Theory - Implementation - Applications, Springer, Boston, 1996.
[127] I.D. Landau, R. Lozano, M. M'Saad, A. Karimi, Adaptive Control: Algorithms, Analysis and Applications, Springer, London, 2011.
[128] D. Antonucci, et al., Building performance evaluation through a novel feature selection algorithm for automated arx model identification procedures, Energy Build. 150 (2017) 432–446.
[129] L.M. Candanedo, V. Feldheim, D. Deramaix, Data driven prediction models of energy use of appliances in a low-energy house, Energy Build. (2017).
[130] I. Naveros, C. Ghiaus, Order selection of thermal models by frequency analysis of measurements for building energy efficiency estimation, Appl. Energy 139 (2015) 230–244.
[131] Y. Chen, A.K. Athienitis, K. Galal, Modeling, design and thermal performance of a BIPV/T system thermally coupled with a ventilated concrete slab in a low energy solar house: part 1, BIPV/T system and house energy concept, Sol. Energy 84 (2010) 1892–1907.
[132] Y. Chen, K. Galal, A.K. Athienitis, Modeling, design and thermal performance of a BIPV/T system thermally coupled with a ventilated concrete slab in a low energy solar house: part 2, ventilated concrete slab, Sol. Energy 84 (2010) 1908–1919.