Available online at www.sciencedirect.com
ScienceDirect
Building and Environment 43 (2008) 170–180
BUILDING AND ENVIRONMENT
www.elsevier.com/locate/buildenv

Identification of the main thermal characteristics of building components using MATLAB

M.J. Jiménez[a,*], H. Madsen[b], K.K. Andersen[b]

[a]CIEMAT, Department of Energy, Energy Efficiency in Buildings Unit, Av. Complutense 22, E28040 Madrid, Spain
[b]Informatics and Mathematical Modelling, Technical University of Denmark, Building 321, DK 2800 Lyngby, Denmark

Abstract

This paper presents the application of the IDENT Graphical User Interface of MATLAB to estimate the thermal properties of building components from outdoor dynamic testing, imposing appropriate physical constraints and assuming linear and time invariant parametric models. The theory is briefly described to provide the background for a first understanding of the models used. The relationship between commonly used RC-network models and the parametric models proposed is presented. The analysis is generalised for different possibilities in the assignment of inputs and outputs and even multiple output. Step by step guidance illustrated by an example is included. Results obtained using the different possibilities in selecting inputs and outputs are reported.

Keywords: Building energy; Thermal parameters; Outdoor testing; System identification

1. Introduction

Thermal modelling of buildings has important applications such as designing control strategies and estimating parameters characterising the system. The latter is of special importance when the parameters of interest have physical meaning, i.e. the estimates reflect physical quantities such as resistances, capacities and time constants of the building. There are a variety of tools for estimating parameters of buildings, for example see [1,2]. These tools differ in applied estimation method, user interface and, in particular, the kind of model types that can be applied. Most of these tools are based on deterministic differential equations or state space models.

The purpose of this paper is to describe how the MATLAB System Identification (IDENT) toolbox [3], together with appropriate physical constraints, can be used to estimate such physical parameters. The IDENT toolbox allows for estimation of very general ARMAX models. This model class is often associated with black-box models and has traditionally been used in prediction, simulation and control applications but very few references report its use to estimate thermal parameters of building components. This paper demonstrates how the model parameters can be given physical interpretation by comparing the ARMAX models with equivalent physical models. Furthermore, the IDENT toolbox allows for both estimation of simulation and prediction-based models, the former being the most commonly used method within thermal modelling for simulation purposes, whereas the latter should be applied whenever the values of physical parameters are of interest.

The paper is organised as follows. First, a mathematical description of linear and time invariant models is given. A special case of this model class is the ARMAX model [4], which can be used for estimating physical parameters of thermal systems. In Section 3, the formulation of classical RC-networks of thermal systems as ARMAX models is described. Subsequently, the physical interpretation of the parameters of the resulting ARMAX model is explained. The method is illustrated by considering a simple RC network model for the heat dynamics of a wall. Section 4 briefly describes the two different estimation methods, the output error method and the prediction error method. The former is relevant when the model is intended for

*Corresponding author. Tel.: +34 950 387922; fax: +34 950 365015.
E-mail address: mjose.jimenez@psa.es (M.J. Jiménez).

0360-1323/$ - see front matter
doi:10.1016/j.buildenv.2006.10.030

Nomenclature

U	heat transmission coefficient, $W\,m^{-2}\,K^{-1}$
g	solar energy transmittance
ϕ	heat flux through the building component, W
Q	heat flux density through the building component, $W\,m^{-2}$
G_v	vertical global irradiance, $W\,m^{-2}$
T_i	indoor air temperature, °C
T_e	outdoor air temperature, °C
ΔT	temperature difference between indoor and outdoor air, °C
A	area of the tested building component, m^2

simulation purposes, whereas the latter is relevant for control (or prediction) purposes, or when the physical parameters are of interest, e.g. in fault detection and diagnosis. Section 6 gives a brief discussion on validation methods, an integral part of model building. A case study is presented in Section 7 to illustrate the use of the MATLAB IDENT toolbox.

2. Linear and time-invariant (LTI) models

This section gives a mathematical description of linear and time-invariant models with input and output signals (input–output systems, for example RC-networks). The models in the IDENT toolbox belong to the family of linear and time-invariant models.

2.1. Non-parametric models

2.1.1. Time domain

For linear and time-invariant systems there exists a function called the *impulse response* function h_k, which describes the relationship between the input signal u_t and the output signal y_t. For causal single-input-single-output (SISO) systems the relationship is described by the folding sum[1]:

$$y_t = \sum_{k=0}^{\infty} h_k u_{t-k}. \tag{1}$$

Using the back-shift operator,[2] defined by $q^{-1}y_t = y_{t-1}$, and a generalised operator function $h(q)$ (often called the *transfer* function)

$$h(q) = h_0 + h_1 q^{-1} + h_2 q^{-2} + \ldots. \tag{2}$$

the relationship between input and output can be written as

$$y_t = h(q)u_t. \tag{3}$$

In practise, some deviations between $h(q)u_t$ and the measured output y_t must be seen. These deviations are described by introducing a noise term N_t, i.e.

$$y_t = h(q)u_t + N_t. \tag{4}$$

The noise term N_t is in general autocorrelated noise.

[1]Notation in this document assumes sampling interval as one time unit, which simplifies notation without losing generality. For example in (1) if the sampling interval is Δt then u_{t-k} should be replaced by $u_{t-k\Delta t}$.

[2]If the sampling interval is Δt, then the back-shift operator is defined by $q^{-1}y_t = y_{t-\Delta t}$.

In the multiple-input–multiple-output (MIMO) case, $h(q)$ is a matrix containing the individual scalar impulse response functions between the corresponding elements of the vectors u_t and y_t.

Using the summation operator $S(q) = (1 + q^{-1} + q^{-2} + \ldots)$ the *step response* function S_k is easily obtained from the impulse response function

$$S_k = S(q)h_k. \tag{5}$$

The impulse response function acts as the most flexible description of linear time-invariant systems (lumped parameter models are a subset of such systems).

Steady-state relations between input u_t and output y_t are obtained by putting $q = 1$, which is equivalent to considering y_t and u_t as constants.

2.1.2. Frequency domain

In system identification the use of frequency domain descriptions is very popular. In this document, however, only a short description of frequency domain methods is provided.

The description in the frequency domain is simply obtained by using the Fourier transformation, and it is easily shown that the fundamental folding (see Eq. (1)), leads to a simple multiplication in the frequency domain

$$Y(\omega) = H(\omega)U(\omega) + N(\omega), \tag{6}$$

where Y, U and N are the signals in the frequency domain, and H is the *frequency response* function (sometimes, but misleadingly, called the transfer function).

Hence, the frequency response function is simply obtained by a Fourier transformation of the impulse response function

$$H(\omega) = \sum_{-\infty}^{\infty} h_k e^{-i\omega k}. \tag{7}$$

As the name indicates, the frequency response function describes how the system reacts to single frequencies. This is often useful in thermal modelling, for example in modelling how the wall reacts to variations with a frequency corresponding to the diurnal cycle.

The frequency response function is a complex function and the modulus is called the gain from input to output, and the argument is the phase-shift from input to output.

Finally it should be noted that *steady-state* relations are obtained at zero frequency (i.e. for $\omega = 0$).

2.2. Parametric models

As described previously the impulse response function provides the most flexible description. The number of parameters is, however, unlimited. In many situations the system can be described (or approximated) using a limited number of parameters.

Most conveniently the relationship is described by models where the transfer function can be described as a rational function, viz.

$$h(q) = \frac{B(q)q^{-b}}{A(q)}, \tag{8}$$

where $A(q)$ and $B(q)$ are polynomials in the back-shift operator

$$A(q) = 1 + a_1 q^{-1} + \cdots + a_r q^{-r}, \tag{9}$$

$$B(q) = b_0 + b_1 q^{-1} + \cdots + b_s q^{-s} \tag{10}$$

and b is an integer-valued time delay from input to output.

Now the parametric equivalent to Eq. (4) is obtained as follows:

$$y_t = \frac{B(q)}{A_1(q)} q^{-b} u_t + N_t. \tag{11}$$

By further specifying an ARMA model for N_t as

$$N_t = \frac{C(q)}{A_2(q)} \varepsilon_t, \tag{12}$$

where ε_t is a sequence of uncorrelated random variables (a white noise process) and $C(q)$ is a polynomial in the back-shift operator, the *Box-Jenkins transfer* function *model* is obtained (Eq. (13)).

$$y_t = \frac{B(q)}{A_1(q)} q^{-b} u_t + \frac{C(q)}{A_2(q)} \varepsilon_t. \tag{13}$$

The term $B(q)q^{-b}/A_1(q)$ is called the *transfer* function *component*.

In modelling it is often assumed that $A_1(q) = A_2(q) = A(q)$, whereby the *ARMAX-model* (Auto-Regressive-Moving Average with eXogeneous input) is obtained (Eq. (14)).

$$A(q)y_t = B(q)u_{t-b} + C(q)\varepsilon_t, \tag{14}$$

where $A(q)$, $B(q)$ and $C(q)$ are pth, sth and qth order polynomials, respectively, in the shift operator q.

Special cases of the ARMAX model are the ARX-model ($C(q) = 1$) and finite impulse response (FIR) ($A_2(q) = C(q) = 1$). The ARMAX model is sometimes referred to as the CARMA model (Controlled ARMA).

An *Output Error Model* (OE model) is a model of the form

$$y_t = \frac{B(q)}{A(q)} u_{t-b} + N_t, \tag{15}$$

where there is no model for the noise term. These models are most often applied in estimation by the *Output Error Method* (OEM). This will be elaborated in Section 4 on estimation.

It is straightforward to extend the model to the case of multiple inputs, and relatively straightforward also to the multiple input–multiple output case.

3. From RC-network to ARMAX models

In this section, it will be shown how physical linear models in state space forms can be formulated as ARMAX models. The link between state space and ARMAX models will be demonstrated.

Consider a wall mounted in a test cell for outdoor testing as sketched in Fig. 1. The wall consists of an outer wall, an insulating part and an inner wall represented as thermal network model (Fig. 2).

T_e is the outside temperature, T_i is the inside temperature, Q is the heat flux density through the wall (measured on the inner surface), and G_v is the global solar irradiance on the outdoor surface of the wall.

The heat exchange through the wall is induced by the temperature difference between indoors and outdoors and by the solar irradiance incident on the outdoor surface (Fig. 1). The thermal characteristics of the wall governing the heat flux rate crossing it are the heat transmission coefficient U and the solar energy transmittance g, as defined by the following steady-state equation:

$$Q = U(T_i - T_e) - gG_v. \tag{16}$$

For dynamic conditions the heat capacitance C, should also be considered. A simple dynamic model for the temperature variations of the wall can be used to estimate the heat conductance and heat capacitance of the various components of the wall. The RC diagram shown in Fig. 2 represents a frequently used model for this kind of wall.

H_1, H_2 and H_3 are the heat conductances corresponding to the outer wall, the insulating part and the inner wall respectively. C_1 and C_2 are the heat capacitances of the outer wall and the inner wall. T_1 and T_2 are the temperature of the outer wall and the inner wall, respectively. (In this case, the conductances and capacitances are considered per unit area.) U can be obtained from the relationship $1/U = 1/H_1 + 1/H_2 + 1/H_3$.

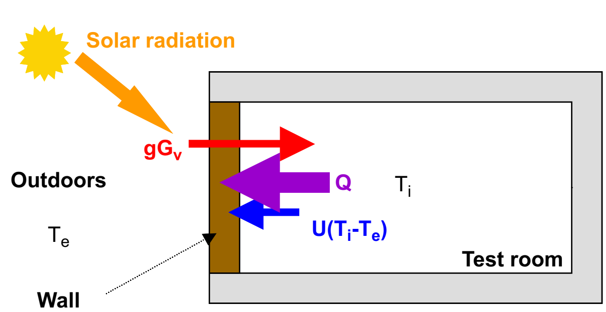

Fig. 1. Scheme of wall installed in a PASLINK test cell and main heat fluxes crossing it.

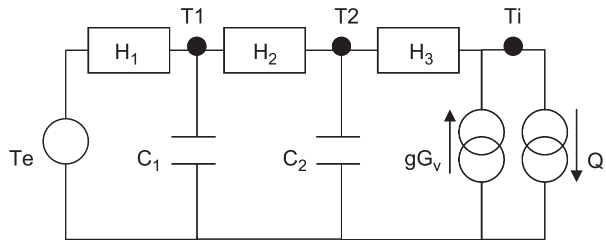

Fig. 2. A simple RC network model for the heat dynamics of a wall.

For this system different possibilities for selecting the output (or response variable) exist. Some analysis techniques applied for this type of test were reported in [5,6]. Most traditional approaches used either the inside temperature T_i or the heat flux density Q as the output variable. However, both may be used together as output variables. The possibilities for arrangement of inputs and outputs, using single-output and multi-output models, are described below.

3.1. Using Q as output variable

The dynamic state space model for this system can be written as

$$C_1\frac{dT_1}{dt} = H_1(T_e - T_1) + H_2(T_2 - T_1) + \frac{d\omega_1}{dt}, \tag{17}$$

$$C_2\frac{dT_2}{dt} = H_2(T_1 - T_2) + H_3(T_i - T_2) + \frac{d\omega_2}{dt}, \tag{18}$$

$$Q = H_3(T_i - T_2) + e; \quad e \in N(0, \sigma_q^2). \tag{19}$$

Three variables are measured, namely T_e, T_i and Q, with measurement errors, which are presented in the dynamic equations as $(d\omega_1, d\omega_2)^T$ and e, respectively. e is a white noise process, while ω_1 and ω_2 are two independent Wiener processes with incremental standard deviations σ_1 and σ_2.

Rearranging terms in Eqs. (17)–(19) leads to

$$\begin{bmatrix} dT_1 \\ dT_2 \end{bmatrix} = \begin{bmatrix} -\frac{1}{C_1}(H_1 + H_2) & \frac{H_2}{C_1} \\ \frac{H_2}{C_2} & -\frac{1}{C_2}(H_2 + H_3) \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} dt + \begin{bmatrix} \frac{H_1}{C_1} & 0 \\ 0 & \frac{H_3}{C_2} \end{bmatrix} \begin{bmatrix} T_e \\ T_i \end{bmatrix} dt + \begin{bmatrix} d\omega_1 \\ d\omega_2 \end{bmatrix}, \tag{20}$$

$$Q = H_3(T_i - T_2) + e; \quad e \in N(0, \sigma_Q^2). \tag{21}$$

Introducing a_{ij} and b_{ij} to simplify notation and considering only the deterministic part of the state space model, Eq. (20) can be written as

$$\begin{bmatrix} dT_1 \\ dT_2 \end{bmatrix} = \begin{bmatrix} -a_{11} & a_{12} \\ a_{21} & -a_{22} \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} dt + \begin{bmatrix} b_{11} & 0 \\ 0 & b_{22} \end{bmatrix} \begin{bmatrix} T_e \\ T_i \end{bmatrix} dt. \tag{22}$$

Taking the Laplace transform gives

$$\begin{bmatrix} s + a_{11} & -a_{12} \\ -a_{21} & s + a_{22} \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} = \begin{bmatrix} b_{11} & 0 \\ 0 & b_{22} \end{bmatrix} \begin{bmatrix} T_e \\ T_i \end{bmatrix} \tag{23}$$

and by isolating T_1 and T_2

$$\begin{bmatrix} T_1 \\ T_2 \end{bmatrix} = \frac{\begin{bmatrix} b_{11}(s + a_{22}) & a_{12}b_{22} \\ a_{21}b_{11} & b_{22}(s + a_{11}) \end{bmatrix} \begin{bmatrix} T_e \\ T_i \end{bmatrix}}{\det A}. \tag{24}$$

The measurement equation equivalent to (21) is

$$Q = c(T_i - T_2) + e; \quad e \in N(0, \sigma_Q^2). \tag{25}$$

Substituting (24) into (25) gives

$$Q = cT_i - c\frac{a_{21}b_{11}T_e + b_{22}(s + a_{11})T_i}{\det A} + e \tag{26}$$

which is an ARMAX model of the form

$$A(q)Q = B_1(q)T_e + B_2(q)T_i + C(q)e. \tag{27}$$

If the solar radiation, G_v, was also available then the model could easily be extended. This extension allows for a simultaneous estimation of the g value, as shown later in Section 5.

Using the RC-network as the starting point, the order of the polynomials follows directly from the specified RC-network; however using ARMAX models it is very easy to try out different orders of the polynomials. The heat equation for such a thermal system is a complex equation that can be characterised by a high number of eigenvalues: the considered RC model is only a simple approximation of this complexity.

3.2. Using T_i as output variable

It is well known that instead of using Q as the output variable, T_i can be selected as the output variable. In this case the model becomes

$$A(q)T_i = B_1(q)T_e + B_2(q)Q + C(q)e. \tag{28}$$

Finally, assuming that measurements of solar radiation G_v are available, the ARMAX model is

$$A(q)T_i = B_1(q)T_e + B_2(q)q + B_3(q)G_v + C(q)e. \tag{29}$$

3.3. Using both T_i and Q as output variables

If both T_i and Q are selected as output variables, the following multiple ARMAX model[3] is obtained:

$$\begin{bmatrix} A_{11}(q) & A_{12}(q) \\ A_{21}(q) & A_{22}(q) \end{bmatrix} \begin{bmatrix} T_i \\ Q \end{bmatrix} = \begin{bmatrix} B_{11}(q) & B_{12}(q) \\ B_{21}(q) & B_{22}(q) \end{bmatrix} \begin{bmatrix} T_e \\ G_v \end{bmatrix} + \begin{bmatrix} C_{11}(q) \\ C_{21}(q) \end{bmatrix} e, \tag{30}$$

[3]The state space model in Fig. 2 would need some minor modifications.

where $A_{ij}(q)$ and $B_{ij}(q)$ are polynomials of appropriate order.

Another possibility is the following multiple ARMAX model:

$$\begin{bmatrix} A_{11}(q) & A_{12}(q) \\ A_{21}(q) & A_{22}(q) \end{bmatrix} \begin{bmatrix} T_i - T_e \\ Q \end{bmatrix} = \begin{bmatrix} B_{11}(q) \\ B_{21}(q) \end{bmatrix} G_v + \begin{bmatrix} C_{11}(q) \\ C_{21}(q) \end{bmatrix} e. \tag{31}$$

4. Estimation methods

In this paper only the *Output Error Method—OEM* and the *Prediction Error Method—PEM* will be described briefly. In the literature a lot of other methods can be found. In the MATLAB IDENT toolbox both OEM and PEM methods are provided. PEM has been used later on in this work.

4.1. OEM

By applying the OEM the unknown parameters (denoted θ) are estimated by

$$\hat{\theta} = \arg\min_{\theta} \left\{ S(\theta) = \sum_{t=1}^{N} N_t^2(\theta) \right\}, \tag{32}$$

where

$$N_t(\theta) = y_t - \frac{B(q)}{A_1(q)} q^{-b} u_t. \tag{33}$$

$N_t(\theta)$ is the *simulation error*, since it represents the deviation between y_t and the simulated output from the model.

The covariance matrix for the parameter estimates is often very difficult to calculate using the output error method. Furthermore, the most common validation techniques cannot be applied, e.g. test for white noise and normally distributed residuals. Therefore, the method should only be used under special conditions.

4.2. Prediction error methods

The prediction error estimate is found by minimising the sum of the squared prediction errors

$$\hat{\theta} = \arg\min_{\theta} \left\{ S(\theta) = \sum_{t=1}^{N} \varepsilon_t^2(\theta) \right\}, \tag{34}$$

where

$$\varepsilon_t(\theta) = Y_t - \hat{Y}_{t/t-1}(\theta) \tag{35}$$

is the one-step prediction error or residual.

If the predictions are calculated using the correct model, then the one-step prediction error becomes a white noise sequence. Using various tests for white noise this enables a powerful possibility for model validation.

The covariance matrix for the parameters estimates is readily obtained using the prediction error method. Furthermore, the method provides an estimation of a model for the errors. The parameters belonging to the C-polynomial can be used, for instance, to obtain an error decomposition; more precisely the total error can be divided into measurement error and model error. This fact enables a powerful tool for model evaluation.

5. Obtaining the physical parameters

In Section 3, it was shown how to obtain an ARMAX model for a specified RC-network model (or state space model) as traditionally used for identification of the heat dynamics of building components. A description is now given of how to obtain the physical parameters (e.g. U-value, g-value, time constants, etc.) using ARMAX models.

As an example the following ARMAX model is considered:

$$A(q)T_i = B_1(q)T_e + B_2(q)Q + B_3(q)G_v + C(q)e. \tag{36}$$

5.1. U value

The well known steady-state relation is

$$Q = U(T_i - T_e) - gG_v + e \tag{37}$$

that must coincide with the steady-state equation obtained from (36) by putting $q = 1$. The following two estimates of the U-values are obtained by comparing these two equations:

$$U_1 = \frac{A(1)}{B_2(1)}, \tag{38}$$

$$U_2 = \frac{B_1(1)}{B_2(1)}. \tag{39}$$

Note that

$$A(1) = \left(1 + \sum_{i=1}^{r} a_i\right), \quad B_1(1) = \left(\sum_{i=0}^{s} b_{1i}\right),$$
$$B_2(1) = \left(\sum_{i=0}^{s} b_{2i}\right), \quad B_3(1) = \left(\sum_{i=0}^{s} b_{3i}\right). \tag{40}$$

In each case the U value is a function of the parameters

$$U_i = f_i(\theta), \tag{41}$$

where $\theta = (a_1, a_2, \ldots, a_r, b_{11}, b_{12}, \ldots b_{1s}, b_{21}, b_{22}, \ldots b_{2s}, b_{31}, b_{32}, \ldots, b_{3s})^T$.

The covariance matrix $V(\theta)$ of the parameter estimates is provided by the MATLAB IDENT toolbox. The variance of the individual U-values can be obtained by the well-known formula for error propagation

$$V(U) = \begin{pmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{pmatrix} = \left(\frac{\partial f}{\partial \theta}\right) V(\theta) \left(\frac{\partial f}{\partial \theta}\right)^T, \tag{42}$$

where

$$f(\theta) = \begin{pmatrix} U_1 \\ U_2 \end{pmatrix}. \quad (43)$$

Therefore

$$\left(\frac{\partial f}{\partial \theta}\right) = \begin{pmatrix} \frac{\partial U_1}{\partial a_1} & \cdots & \frac{\partial U_1}{\partial a_r} & \frac{\partial U_1}{\partial b_{11}} & \cdots & \frac{\partial U_1}{\partial b_{1S}} & \frac{\partial U_1}{\partial b_{21}} & \cdots & \frac{\partial U_1}{\partial b_{2s}} & \frac{\partial U_1}{\partial b_{31}} & \cdots & \frac{\partial U_1}{\partial b_{3s}} \\ \frac{\partial U_2}{\partial a_1} & \cdots & \frac{\partial U_2}{\partial a_r} & \frac{\partial U_2}{\partial b_{11}} & \cdots & \frac{\partial U_2}{\partial b_{1s}} & \frac{\partial U_2}{\partial b_{21}} & \cdots & \frac{\partial U_2}{\partial b_{2s}} & \frac{\partial U_2}{\partial b_{31}} & \cdots & \frac{\partial U_2}{\partial b_{3s}} \end{pmatrix}. \quad (44)$$

The two values estimated for U can be combined into a single value by a Lagrange weighting as

$$U = \lambda U_1 + (1-\lambda)U_2. \quad (45)$$

Using the variance of each of the two U-values to determine the weighting, λ is obtained from the element of the covariance matrix v_{ij} as

$$\lambda = \frac{v_{22} - v_{12}}{v_{11} + v_{22} - 2v_{12}} \quad (46)$$

and the corresponding standard deviation as the square root of its variance as

$$\sigma(U) = \sqrt{\frac{v_{11}\,v_{22} - v_{12}^2}{v_{11} + v_{22} - 2\,v_{12}}}. \quad (47)$$

5.2. *g-value*

The g-value is determined as

$$g = \frac{B_3(1)}{B_2(1)} \quad (48)$$

and the variance of the estimate is again calculated by using the variance propagation approach, as described above.

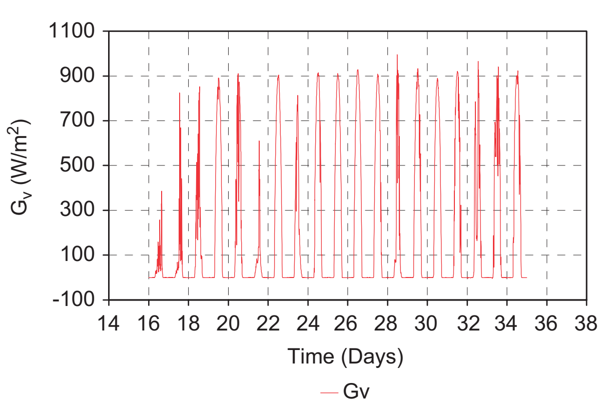

Fig. 3. Global solar irradiance on the external surface of the component . Data set no. 2.

5.3. *Time constants*

The time constants are related to the A-polynomial. The number of time constants is equal to the order of the A-polynomial.

Assuming that the order of the A-polynomial is r then the polynomial $A(q)$ has r roots. For each root p_i the corresponding time constant τ_i is calculated as

$$\tau_i = \frac{1}{\ln(p_i)}. \quad (49)$$

5.4. *Impulse and step response functions*

For each combination of inputs and outputs an impulse response function can be found by a polynomial division of the relevant polynomials of the ARMAX model.

For instance the impulse response function from the solar radiation to the indoor air temperature is calculated as

$$h(q) = \frac{B_3(q)}{A(q)} \quad (50)$$

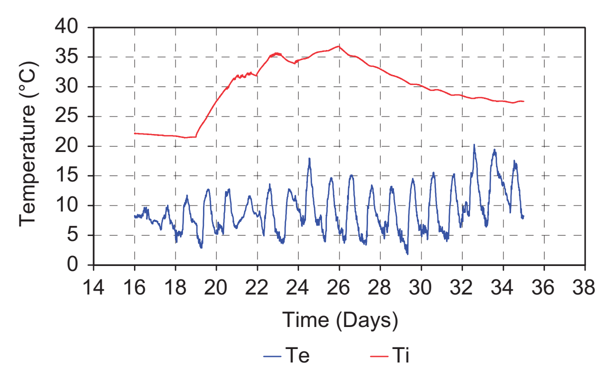

Fig. 4. Indoor and outdoor air temperatures. Data set no. 2.

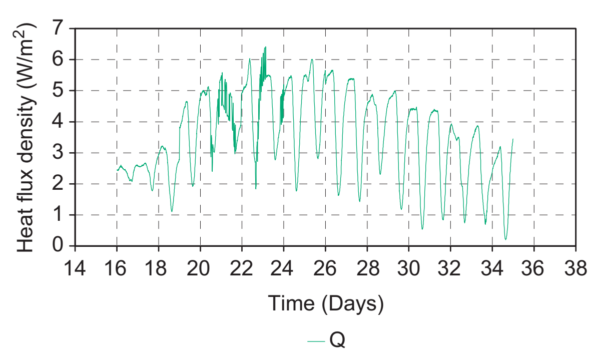

Fig. 5. Heat flux density leaving the test room measured at the inside surface of the component. Data set no. 2.

and the step response function is then found by a summation of the impulse response function, as described by Eq. (5).

6. Model validation

When building a dynamic model it is important that certain criteria for the model fit are met to ensure an adequate model. Several procedures for model validation exist [7]. The following criteria for model determination and validation were suggested by Norlén [8]

1. *Fit to the data*. The model residuals should be 'small' and 'white noise'. A necessary condition for 'white noise' is that there should be neither autocorrelation in the residuals nor correlation between the residuals and the input variables.
2. *Internal validity*. The model should agree with data other than data used for parameter estimation (cross validation).
3. *External validity*. The result from the model should not (without greater motivation) conflict with previous experiences or other known conditions.
4. *Dynamic stability*. From a steady state, the model should give an output upon a temporary change in an input variable that gradually fades out (if the model is intended to describe dynamic characteristics).
5. *Identifiability*. It should be possible to determine the parameters of the model uniquely from the data.
6. *Simplicity*. The model should be as small as possible.

7. Case study

7.1. Objective of the test

As part of a series of round robin tests carried out at several European PASLINK test sites, an opaque component, described below, was tested under dynamic condition in order to determine its U-value [9]. Ref. [9] also provides useful, additional information for external validation.

7.2. Test component

The component is a simple homogeneous, opaque wall. It consists of a sandwich of insulation between plywood,

Table 1
Identified parameters for an ARX model with $na = 10$, $nb = [10\ 10\ 10]$ and $nk = [1\ 1\ 1]$

	a_i	b_{1i}	b_{2i}	b_{3i}
$i = 1$	−5.98E−01	−1.95E−03	7.83E−02	3.13E−05
$i = 2$	−3.00E−01	5.53E−03	−5.18E−02	−6.46E−07
$i = 3$	−2.45E−01	−3.64E−03	8.17E−03	−1.19E−05
$i = 4$	−6.80E−02	1.16E−03	2.35E−02	−2.13E−05
$i = 5$	−1.53E−02	−6.33E−04	−2.82E−02	8.03E−06
$i = 6$	4.75E−02	3.80E−03	−6.65E−03	2.69E−06
$i = 7$	1.70E−02	−3.68E−03	1.37E−02	5.83E−06
$i = 8$	5.77E−02	−3.70E−03	−2.22E−02	2.97E−05
$i = 9$	4.63E−02	4.73E−03	1.08E−02	−1.29E−06
$i = 10$	6.01E−02	1.23E−03	−1.09E−02	6.54E−06

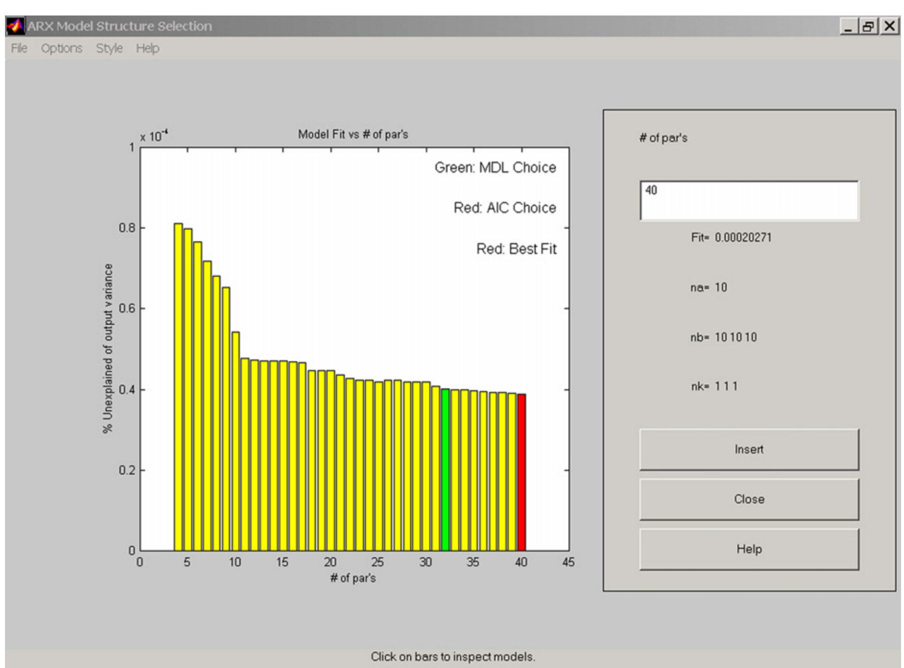

Fig. 6. Results of model selection, using T_e, Q and G_v as inputs, and Ti as output.

Table 2
Physical parameters

Parameter	Estimate	Units
U_1	1.86E−01 ± 3.46E−03	W/m^{2}K
U_2	1.94E−01 ± 1.37E−02	W/m^{2}K
g	3.32E−03 ± 7.05E−04	—

Table 3
Results

Parameter	Estimate	Units
U	0.191 ± 0.009	W/m^{2}K
g	0.0033 ± 0.0007	—

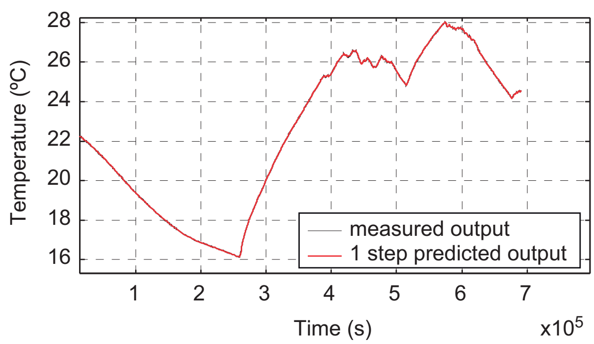

Fig. 7. Measurements and predicted outputs given by the "Best Fit" models versus time.

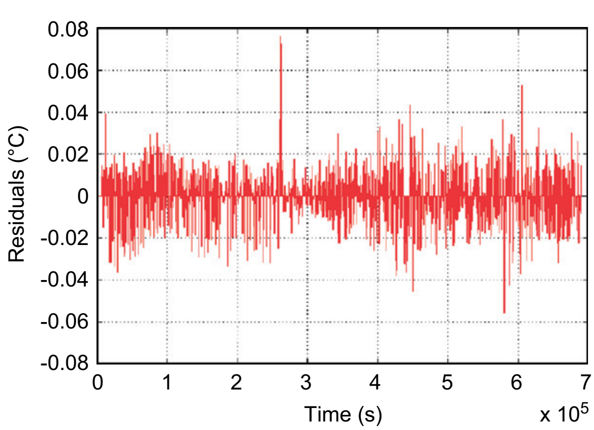

Fig. 8. Residual given by the "Best Fit" models versus time. The average of the residuals is 0.0002 °C.

and the following materials have been used for its construction:

- Two hundred mm thick expanded polystyrene (PS30), density 30 kg/m^3, consisting of 4 layers 50 mm thick.
- Two 12 mm thick sheets of phenolic faced plywood.
- White melamine 0.7 mm thick for outdoor face.
- Polyurethane adhesive to glue layers together.
- Sixteen nylon bolts, washers and nuts size M12, uniformly distributed, to mechanically clamp the sandwich together.

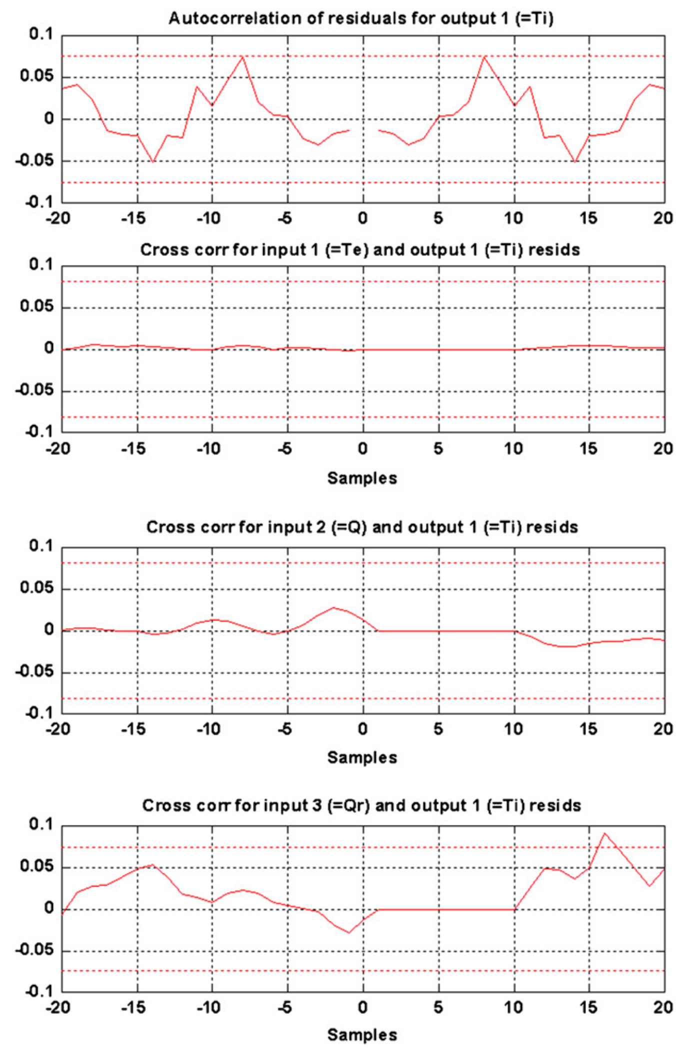

Fig. 9. Residual autocorrelation and cross-correlation.

A more detailed description can be found in [9].

7.3. Test procedure

The component was installed in the aperture of a PASLINK test cell [10]. The following sensors were used for measurements:

- A shielded/aspirated platinum resistance sensor (PRT) for external air temperature.
- Seven shielded PRTs for test room air temperatures.
- A pyranometer for solar irradiance on the component surface.
- A thermopile-based transducer for heat flux density, mounted flush with the inside surface of the component.

Tests were carried out according to the PASLINK test procedure [11]. This procedure consists of a test sequence including 3 days of low power, 1.5 days of high power and 3.5 days according to a pseudo-random on/off power sequence. The maximum power level is determined by the weather conditions at the test site in order to avoid overheating or undercooling of the test cell, whilst maximising the temperature difference between outside and the test room.

7.4. Data

The following data for the opaque component tested at two different test sites have been analysed:

- *Data set no*. 1: Data analysed corresponding to the component tested at BRE (East Kilbride, near Glasgow, Scotland). These data are available at www.paslink.org.
- *Data set no*. 2: Data from the experiments on the component tested at CIEMAT (Tabernas, Almería, Spain). This experiment was carried out in the period 6th–24th of January 2002. Figs. 3–5 show the measured signals.

7.5. Example of estimation of U- and g-values

This section describes "step by step" the estimation of U and g, using IDENT in MATLAB, for data set no. 1. Detailed guidance explaining how each step is done in the IDENT toolbox is included in [12].

Only ARX models have been considered to simplify the analysis without loss of generality. The following types of models have been considered as candidate model to estimate U and g:

- Models including G_v as input. In this case the small value of g can be determined.
- Models that do not include G_v, which suppose the value of g as negligible and therefore cannot be determined by these models.
- Single output models.
- Multi-output models.

Section 7.6 summarises the results obtained for each of the considered models. As only steady-state parameters are required, only plots of residuals, average of residuals and estimated identification errors have been considered in order to compare the performance of the models.

In this section, an ARX model using T_i as output and T_e, Q and G_v as inputs, as described in Section 3.2, has been considered as an example to illustrate how to obtain the U- and g-values.

The MATLAB IDENT toolbox has been used to examine data, select model structure and order, fit and validate the proposed model, and also to generate the matrix containing information about the selected model structure, estimated parameters and their estimated accuracy. In this section, the notation used in MATLAB for the orders of polynomials and delay between input and output, has been used. In this notation *na* represents the order of polynomial $A(q)$, *nb* the order of polynomial $B(q)$, and *nk* the delay between output and input. The study has been limited to models with, *na*, *nb* and *nk*, varying between 1 and 10. The models marked by IDENT as "Best Fit" are those where orders are $na = 10$, $nb = [10\,10\,10]$ and $nk = [1\,1\,1]$. The results of these estimations are presented in Fig. 6.

In this case "Best Fit" models have been selected and used to estimate the required parameters.

The matrix containing information about the selected model structure is then exported to the MATLAB

Table 4
Results obtained for the data set no. 1 using all data for identification and validation: U and g

Model number	Inputs	Outputs	Model	U (W/m^{2}K)	error_U (W/m^{2}K)	g (—)	error_g (—)
1	Q	T_i–T_e	arx10101(BF)	0.19	0.03	N/A	N/A
2	Q,G_v	T_i–T_e	arx10101(BF)	0.21	0.03	0.02	0.01
3	T_e,Q,G_v	T_i	arx10101(BF)	0.191	0.009	0.0033	0.0007
4	T_e,Q	T_i	arx10101(BF)	0.20	0.01	N/A	N/A
5	G_v	T_i–T_e,q	arx221	0.178	0.007	0.001	0.001

Table 5
Results obtained for data set no. 2. (The first 2/3 of data file has been used for identification and the last 1/3 for validation): U and g

Model number	Inputs	Outputs	Models	U (W/m^{2}K)	error_U (W/m^{2}K)	g (—)	error_g (—)
1	Q	T_i-T_e	arx882(BF)	0.173	0.006	N/A	N/A
2	Q,G_v	T_i-T_e	arx461(AIC)	0.180	0.009	0.0008	0.0007
3	T_i-T_e	Q	arx2107(MDL,AIC,BF)	0.176	0.004	N/A	N/A
4	T_i-T_e,G_v	Q	arx225(BF)	0.191	0.003	0.0020	0.0002
5	T_i,T_e,G_v	Q	arx136(MDL,AIC,BF)	0.197	0.005	0.0029	0.0003
6	G_v	T_i-T_e,q	arx111	0.193	0.003	0.0023	0.0002

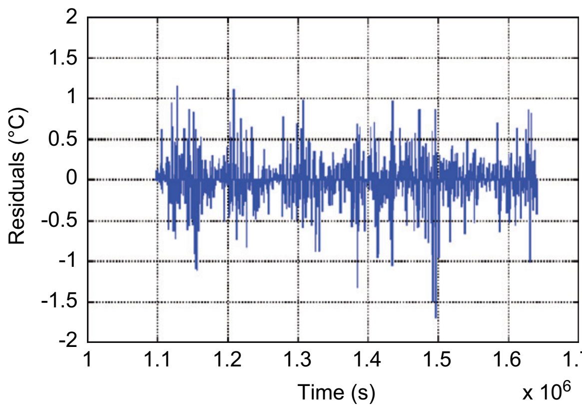

Fig. 10. Residual given by the single output model number 2 of Table 5, versus time. The average of the residuals is −0.01065 °C.

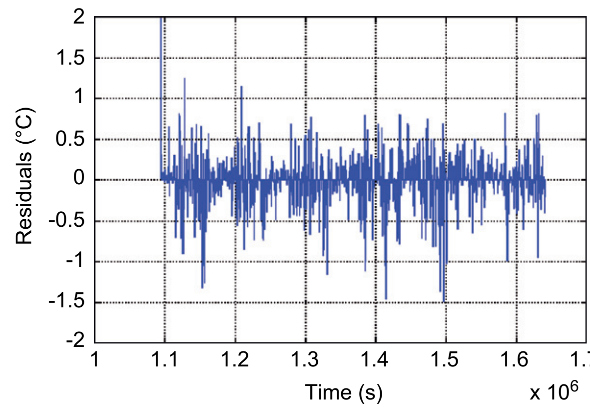

Fig. 12. Residual given by the two outputs model number 6 of Table 5, versus time. The average of the residuals is 0.0002 °C.

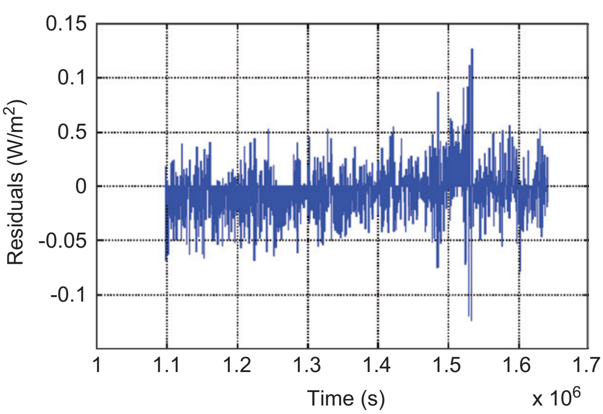

Fig. 11. Residual given by the single output model number 4 of Table 5, versus time. The average of the residuals is $-0.007\,\mathrm{W/m^2}$.

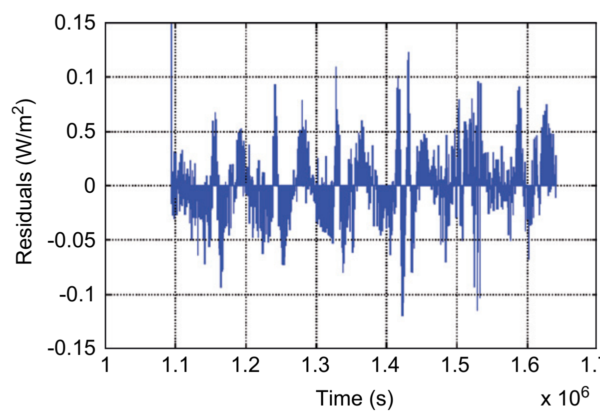

Fig. 13. Residual given by the two outputs model number 6 of Table 5, versus time. The average of the residuals is $0.003\,^{\circ}\mathrm{C\,W/m^2}$.

workspace, where the required parameters and their estimation errors have been obtained.

Table 1 presents the identified parameters for the selected ARX model.

By using (38)–(48) the values in Table 2 were obtained.

Finally, applying Lagrange's method and rounding the result according to the obtained errors the parameter values in Table 3 were obtained.

Afterwards the model was successfully validated using the criteria described in Section 6.

Fig. 7 shows a good fit between measured and predicted output, and Figs. 8 and 9 show low and uncorrelated residuals. External validity is guaranteed as these results agree with the results obtained for the same component tested following the same test procedure at other test sites and also with values estimated from tabulated values. The model has dynamic stability: instability would trigger a warning message by MATLAB. The identifiability of the model is self evident and its simplicity is acceptable.

7.6. Summary of results given by different models

Several models have been tried in both cases and those giving the best results are summarised in Tables 4 and 5 for data set nos. 1 and 2, respectively.

In the models that consider G_v, a very low g-value has been estimated, with relatively high estimated errors. These results are not unexpected for the considered insulating and opaque component, due to the weak influence of G_v in the heat flux density through the component, and for the same reason these errors are not very critical.

Note the low identification error estimated for U-value (Tables 4 and 5), and the low average of residuals (Figs. 10–13) obtained when a two outputs model is considered.

8. Conclusions

In this paper, a procedure for using IDENT in MATLAB for estimating the thermal characteristics of a wall component is suggested. The flexibility of this approach has been demonstrated by the use of several models with different assignment of inputs and outputs and even multiple-outputs models. Very good performance has been found for multiple-output models giving very low identification errors. A different analysis tool (e.g. CTSM [2]) must be applied for those components where non-linear effects are relevant.

References

[1] Gutschker O. Parameter identification with the software package LORD. Building and Environment, this issue.

[2] Kristensen NR, Madsen H. Continuous time stochastic modelling—CTSM 2.3 user guide. Technical University of Denmark, Lyngby, Denmark. 2003. www.imm.dtu.dk/ctsm

[3] Ljung L. System identification. Theory for the user. NJ: Prentice-Hall; 1999.

[4] Box GEP, Jenkins JM. Time series analysis: forecasting and control. San Francisco, USA: Holden-Day; 1976.

[5] SIC 1. Workshop on application of system identification in energy savings in buildings. In: Bloem JJ, editor. Published by the Commission of the European Communities, L-2920 Luxembourg, Ref.: EUR 15566 EN 1994.

[6] SIC 2. System identification competition. In: Bloem JJ, editor. Published by the Office for Official Publications of the European Communities, L-2985 Luxembourg, Ref.: EUR 16359 EN, ISBN 92-827-6348-X 1996.

[7] Madsen H. Time series analysis. Lyngby, Denmark: Institute of Mathematical Modelling, Technical University of Denmark; 2001.

[8] Norlén U. Determining the thermal resistance from in-situ measurements. In: Bloem JJ, editor. SIC1. Workshop on appli cation of system identification in energy savings in buildings. Luxembourg: Published by the Commission of the European Communities, L-2920 Luxembourg, Ref.: EUR 15566 EN 1994; p. 402–29.

[9] Baker PH, Strachan P. WP3 round robin tests as feasibility study for standardization. Final technical report part 3. IQ-TEST Project, Contract no. ERK6-CT1999-2003, 2003.

[10] www.paslink.org, last viewed August 2006.

[11] van Dijk HAL, Tellez FM. Measurement and data analysis procedures. Final report of the JOULE II COMPASS Project (JOU2-CT92-0216), 1995.

[12] Jiménez MJ, Madsen H, Andersen KK. How to get physical parameters using MATLAB. Presented in the conference "international conference on dynamic analysis and modelling techniques". Organized by PASLINK EEIG and JRC. 13–14 November 2003. Proceedings: EUR 21187 EN. ISBN 92-894-7794-6, 2004.