Inverse modeling of buildings with floor heating and cooling systems

*for a real-time benchmark of the operational energy use*

Graduation Project Building Physics and Services (7SS37) – Eindhoven University of Technology

---

**EXISTING OFFICE BUILDING**

**LOW-RESOLUTION INVERSE MODEL**

\[ \begin{align*}
T_i & \quad \bar{T}_i \\
T_{\text{wsup}} & \quad T_{\text{wrec}} \\
\hat{m}_w & \\
\int \quad P & \rightarrow E
\end{align*} \]

- **Weather data**
  - \( T_e \)
  - \( I_{\text{glob}} \)

- **Control settings**
  - \( T_{\text{set}} \)
  - \( T_{\text{wsup}} \)

- **Estimated**
  - \( \hat{T}_i \)
  - \( \hat{E} \)

---

**Student:**
J.M. (Rowan) de Nijs (0738286)

**Supervisors:**
- prof.dr.ir. J.L.M. (Jan) Hensen
- dr.ir. M.G.L.C. (Marcel) Loomans
- dr.ir. J.I. (Ignacio) Torrens Galdiz
- ir. R.P. (Rick) Kramer
- dr. ir. C.J. (Kees) Wisse

**Date:**
07 – 12 – 2015
COLOPHON

Project information
Project title: Inverse modeling of buildings with floor heating and cooling systems for a real-time benchmark of the operational energy use

Course: Graduation Project Building Physics and Services (7SS37)
Graduation period: 01/11/2014 – 15/12/2015

Student information
Name Student: J.M. (Rowan) de Nijs
E-mail: j.m.d.nijs@students.tue.nl
Telephone: +31 (0)6 31 35 85 13
Student Number: 0738286 (S096707)

Study program related information
University: Eindhoven University of Technology (TU/e)
Master program: Architecture, Building and Planning (ABP)
Master track: Building Physics and Services (BPS)
Chair: Computational Building Performance Simulation (CBPS)

Graduation committee

University of Technology Eindhoven
Chairman: prof.dr.ir. J.L.M. (Jan) Hensen
j.hensen@tue.nl

Daily supervisors: dr.ir. M.G.L.C. (Marcel) Loomans
m.g.l.c.loomans@tue.nl

dr.ir. J.I. (Ignacio) Torrens Galdiz
j.i.torrens@tue.nl

ir. R.P. (Rick) Kramer
r.p.kramer@tue.nl

DWA
Daily supervisor: dr. ir. C.J. (Kees) Wisse
wisse@dwa.nl
ABSTRACT

This paper proposes a methodology to build and validate a low-resolution resistance-capacitance (RC) model in combination with inverse modeling for confident estimations of annual heating and cooling demands for buildings with water-based heating and cooling systems. A virtual single-zone office building with a floor heating and cooling system in the high-resolution building simulation tool Vabi Elements is used as case study building. The virtual building is used to identify the parameters of the low-resolution RC model and to validate the model. Both the control strategy and the heating and cooling system (water-based floor system) were modeled as applied in the virtual building. By utilizing multiple case variants, the model’s ability to consider uncertain building parameters was tested. Consequently, enabling confident estimations of expected energy distributions for a real-time benchmark of the operational energy performance in future research steps. The number of parameters in the RC building network was gradually increased until the appropriate model resolution was established. The parameter values were identified using the Matlab Global Optimization Toolbox optimizers Genetic Algorithm, PatternSearch and Fmincon successively. Validation was performed based on a residual analysis and a set of performance metrics.

Keywords: real-time benchmark, inverse modeling, system identification, resistance-capacitance model

1. INTRODUCTION

Within the built environment, the performance of building’s service systems is becoming increasingly important in system maintenance contracts and for Energy Service Companies (ESCOs). Maintenance parties have an increasing stake in this performance, since traditional system maintenance contracts are gradually moving towards performance based contracts. For ESCOs, their associated energy costs cover a significant share of the total exploitation costs of these systems. Therefore, these stakeholders can benefit from insight into the building’s operational energy performance, while considering among others its specific occupancy behavior, weather conditions and system control settings.

Evaluating systems on component level attains insight in the performance of individual system components and their effect on the performance on sub-process levels (bottom-up approach). However, to gain insight into the operational energy use of the entire building system, the system performance should be evaluated on building level. This approach enables to distinguish ‘normal’ and ‘abnormal’ energy use and to diagnose the cause of abnormal energy use on sub-process or component level (top-down).

Due to this emphasis on the total operational energy use, there is an increasing need for a method that evaluates operational energy consumption on whole-building level, taken into account the ‘normal’ range in energy use due to among others occupancy behavior, weather conditions and system control settings. The method should collect on-site measured data and perform a real-time estimation of expected energy distribution to enable continuous insight into the performance. In addition, it should evaluate the operational energy use and distinguish ‘normal’ and ‘faulty’ behavior on small-time scales (hours, days) to enable quick measures for e.g. sudden system faults. Finally, it should evaluate the energy performance on long-time scales (seasons, years) to determine its effect on total energy costs. However, the evaluation of the operational energy use on whole-building level is rather difficult, as it is influenced by many variables and many uncertainties (e.g. weather conditions, building constructions, occupancy behavior, control settings).

In recent years, numerous approaches to the evaluation of operational energy use on whole-building level have been proposed [1]. Evaluation of seasonal and annual energy demands is frequently based on reference values of buildings with similar building characteristics and climate conditions [2]. For the evaluation of (sub-)hourly values, the model-based approaches can be considered. These models can have a ‘black’ box or ‘white’ box structure and be developed using forward or inverse modeling [3].

Black box modeling uses inverse modeling to define a model structure and combination of parameter values that describes the relation between the measured energy consumption and influential variables [4]. Since this structure lacks a physical meaning, it is impossible to utilize the model for other building scenarios which makes them inappropriate for energy distribution estimations. Moreover, these models require a
A comprehensive set of measured data, because the model structure is not prescribed [4]. White box building models have a prescribed model and the parameterization can be performed either forward or inverse, i.e. manually or using system identification principles respectively. It uses physical principles to calculate thermal dynamics and energy behavior [4] and are therefore more effective to consider uncertain parameters. These models generally require more information about the building characteristics as the resolution increases, enhancing the effort for the model development and the calculation time for simulations [5]. Hence, low-resolution physical models are more effective than high-resolution building simulation models (e.g. TRNSys or Vabielements) when deployed for a large number of non-identical buildings and for real-time applications. Moreover, several studies have proven that a low-resolution physical model can already provide the required level of confidence for the purpose of operational energy performance estimations [6–8].

Both parameterizing approaches are applicable for low-resolution physical building models. When comparing the two approaches, the effort for model development and the amount of uncertainties in building parameters is considerably lower for the inverse modeling approach, because the model is then calibrated by an algorithm instead of manually [4]. Research interests in low-resolution models in combination with inverse modeling focus on analogous electrical resistance-capacitance (RC) networks, which due to its flexible structure allows to select the appropriate level of abstraction for the research’s purpose [3], [4], [9–11]. To allow identification of the network parameters, these models consist of only time-invariant parameters. Furthermore, to enable acceptable calculation demands for the inverse modeling process, these networks are modeled in the state-space representation [12]. Consequently, the RC models are linear models, which might complicate the modeling of non-linear physical processes. A number of these processes can be considered as non-linear by preprocessing the input parameters (e.g. solar irradiation). However, some processes must be simulated as linear (wind-dependent infiltration). Nevertheless, several studies demonstrated that this approach is promising to be applied as operational energy model of a building in order to estimate expected energy performance [13–15], even though the entire building envelope is modelled as one ‘lumped’ capacitance [16–18].

A procedure that seems to have potential for the application of lumped RC models in combination with inverse modeling for the evaluation of the operational energy use was proposed by Henze et al. (2015) [19]. This study used a fictive retail building with a packaged all-air Heating Cooling and Air Conditioning (HVAC) system as case study building to demonstrate a four-stage approach: first, a lumped RC building model was developed by using inverse modeling and coupled to an HVAC model; second, the uncertain parameters were quantified (mean value and standard deviation of 10%) as well as the ‘fault’ parameters (e.g. temperature set points, time schedules, ventilation outdoor air fraction); third,
possible ranges in operational energy consumption were estimated by varying the uncertain parameters; fourth, the expected energy use was estimated and ‘normal’ and ‘faulty’ energy use were distinguished.

In contrary to this case study building, many Dutch office buildings have water-based heating and cooling system, which have other heat dynamics, uncertainties and faulty behavior than all-air systems. The modeling of these systems differs as well, due to the considerably higher radiation part in the total delivered power by this system compared to an all-air system. A proof of principle that this procedure is also applicable for buildings with water-based heating and cooling systems has not been established.

For the first stage of Henze’s approach, several studies proposed a methodology for single-zone buildings with water-based systems, but these studies only addressed residential buildings. RC models were built and validated for control strategy evaluations [16], for model predictive control [20], for the application in a boiler control scheme [21] and for the characterization of heat dynamics [22]. Therefore, this research proposes a methodology to build and validate a lumped RC model using inverse modeling for the estimation of heating and cooling demands for office buildings heated and chilled with all-water systems.

The main objective of this research is to develop a lumped RC model that can estimate heating and cooling demands of single-zone office buildings with water-based floor heating and cooling systems. To prove that the model can also be applied in future research steps to estimate expected energy distribution due to uncertain building parameters, the model should correctly estimate the effect of these parameters on heating and cooling demands. Since indoor air temperatures strongly affect the heating and cooling demands, these estimated demands are considered as confident if their associated indoor air temperatures are estimated correctly. The research question was defined as:

When using the inverse modeling approach in combination with a lumped RC model, what model resolution is required to confidently estimate annual heating and cooling demand of a single-zone office building with a floor heating and cooling system?

The outline of the paper is as follows: chapter 2 describes the methodology, which consists of five major steps as shown in figure 1.1. Chapter 3 shows the results of all five steps, while each step is discussed successively in chapter 4. Conclusions and recommendations can be found in chapter 5. The appendix provides additional information.

2. METHODOLOGY

A virtual case study building was selected, which enabled the evaluation of multiple building variants. Initially, a simulation model was developed in a high-resolution building simulation tool to emulate five case variants for this building (a). According to an iterative process of gradually increasing the lumped RC building model resolution, steps b) – e) were executed until an RC building model was found for which the developed model satisfied the validation criteria for all five case variants. Here, it was assumed that the level of confidence of the coupled simulation model depended only on the lumped RC building model resolution. Each turn the model did not satisfy the validation criteria, an RC network was created (b), its parameters were identified (c), the identified network was coupled to a floor system model (d) and the coupled model was validated (e).

Using this methodology, the lowest resolution model was found that complied for the case study building.

a) A typical Dutch office building with a floor heating and cooling system was defined and the uncertain parameters of this building were selected and quantified. Based on this building and these parameters, five case variants were emulated, as described in section 2.1.

b) For this case study building, a lumped RC building model configuration was defined. Starting with the lowest model resolution as clarified in section 2.2, the resolution was gradually increased.

c) For the RC building model configuration of step b), the parameters were identified by using the inverse modeling approach as explained in section 2.3.

d) By coupling the identified building model of step c) to a floor system model and control, indoor air temperatures and heating and cooling demands were estimated, as clarified in section 2.4.

e) The ability of the coupled model of step c) to estimate hourly indoor air temperatures and annual heating and cooling demands for all five case variants was tested based on a set of validation criteria, which are described in section 2.5.
Two artificial sets of climate data with a time interval of 3600 s were used for this study, NEN 5060 E and NEN 5060 ref. TO5 [23]. Both datasets were used for step a), the first dataset was used for the development of the low-resolution simulation model (step b - d), the second dataset for the model validation (step e). Figure 2.1 shows the ambient temperature and global horizontal irradiation for a winter and summer week of the first dataset. Appendix I lists all properties of the two data sets.

2.1 Case study description

A modified version of BESTEST case 900 (high-weight building) was used as case study building, as shown by figure 2.2 [24]. Among others, the building was rotated (windows are oriented to the west), its air heating and cooling system was replaced by a water-based floor system and the ventilation rate of the building was set to zero.

![Figure 2.2: Virtual single-zone case study building (modified BESTEST 900 case) [24]](image_url)

The walls, windows and roof are adjacent to the ambient air, while the floor is connected to the ground with a fixed temperature of 10 °C. The building is located in an open space and the building lacks sun blinds. Table 2.1 shows the main properties of the building constructions. Further details about the case study building can be found in appendix II.

### Table 2.1: Main properties of building construction

<table>
<thead>
<tr>
<th>Construction</th>
<th>( A ) ([\text{m}^2])</th>
<th>( R ) ([\text{m}^2\cdot\text{K}\cdot\text{W}^{-1}])</th>
<th>( \rho A ) ([\text{kg} \cdot \text{m}^2])</th>
<th>( C ) ([\text{J} \cdot \text{K}^{-1}])</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roof</td>
<td>48.0</td>
<td>3.5</td>
<td>21.1</td>
<td>0.9 \times 10^6</td>
</tr>
<tr>
<td>Floor</td>
<td>48.0</td>
<td>25.2</td>
<td>122.1</td>
<td>5.4 \times 10^6</td>
</tr>
<tr>
<td>Walls</td>
<td>54.6</td>
<td>3.5</td>
<td>146.1</td>
<td>8.0 \times 10^6</td>
</tr>
<tr>
<td>Windows</td>
<td>12.0</td>
<td>0.9</td>
<td>23.4</td>
<td>0.2 \times 10^6</td>
</tr>
</tbody>
</table>

Variations were defined in four building parameters of the virtual case study building, as shown by table 2.2. Case variant 1 was the base case scenario, from which case variants 2 - 5 differ by one parameter. The wind-dependent infiltration rate \( n_{\text{inf}} \) of case variant 2 was doubled, to consider the influence of linearizing this uncertain non-linear physical phenomena on the model’s level of confidence. The windows of case variant 3 had a g-value \( g_{\text{win}} \) that is 73 % higher, to analyze the effect of the uncertainty in solar gains, e.g. due to sun blinds or shading.

### Table 2.2: Characteristics of 5 building variants

<table>
<thead>
<tr>
<th>Case variant</th>
<th>Parameter value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( n_{\text{inf}} ) 0.1 h(^{-1}) for ( v = 0 ) m/s 0.5 h(^{-1}) for ( v = 6 ) m/s</td>
</tr>
<tr>
<td>2</td>
<td>( g_{\text{win}} ) 0.400 [-]</td>
</tr>
<tr>
<td>3</td>
<td>( Q_{\text{int}} ) 12 W·m(^{-2}) for ( v = 0 ) m/s 5 W·m(^{-2}) (10 - 18h weekdays)</td>
</tr>
<tr>
<td>4</td>
<td>( T_{\text{iset}} ) 20 – 24 °C (10 - 18h weekdays) 18 – 26 °C (other)</td>
</tr>
<tr>
<td>5</td>
<td>( T_{\text{iset}} ) 21 – 23 °C (10 - 18h weekdays) 19 – 25 °C (other)</td>
</tr>
</tbody>
</table>

![Figure 2.1: Ambient temperature and global horizontal irradiance for a winter (left) and summer (right) week of the NEN 5060 E dataset](image_url)
Case variants 4 and 5 considered the uncertainty in occupancy behavior and system control settings, by doubling the internal heat gains $Q_{\text{int}}$ and applying different temperature set points $T_{\text{set}}$ respectively.

Since variants 2 and 3 differ by one building construction parameter compared to the base case variant, it was necessary to develop a separate lumped RC building model for these variants as shown in table 2.3. The methodology as displayed in figure 1.1 was deployed to develop these three models. The lumped RC building network defined in step b) was used in step c) and d) to identify a separate model for case variants 1 - 3. Subsequently, these three models were individually validated in step e). In contrary, case variant 4 and 5 did not require a separate lumped RC building model, because these variants did not differ by a building construction parameter. Therefore, the developed model of case variant 1 was tested on variants 1, 4 and 5.

The high-resolution building simulation tool Vabi Elements 1.5.1 [25], that meets a.o. the BESTEST [26] and the ASHRAE standard 140 [27], was used to emulate the five case variants. The virtual building model is considered as a representation of an existing version of the case study building. From its floor system, the supply water temperature $T_{\text{w, sup}}$ and flow rate $m_w$ are known control settings ($T_{\text{w, sup}}$ from a heating curve, $m_w = 600$ kg·h$^{-1}$), while the return water temperature $T_{\text{w, ret}}$ is continuously monitored with a time interval of 3600 s. The heating and cooling power was determined from these data and the indoor air temperature was monitored as well with this time interval. The emulated data is presented in section 3.1 and appendix III.

### 2.2 Lumped RC building model

The determination of a lumped RC building network consists of the selection of the inputs and outputs to describe the significant physical phenomena and the definition of the network configuration.

#### Selection of the significant inputs and outputs

Numerous studies showed that indoor air temperatures can be confidently estimated by considering the five physical phenomena displayed in figure 2.3 [21, 28–32].

![Figure 2.3: Physical phenomena considered by the lumped RC building model](image)

When using an RC model, these physical relations are represented as a thermal network consisting of time-invariant resistances and capacitances. Based on this network, a numerical description of the model can be defined by describing an ordinary differential equation (ODE) for each capacitance node. Then, the set of ODEs was defined in the state-space representation, according to eq. (1) and (2), e.g. [12]:

$$\dot{x}_k(t) = Ax_k(t) + Bu_k(t)$$

$$y_k(t) = Cx_k(t) + Du_k(t)$$

In the state-space representation, $A$, $B$, $C$ and $D$ are system matrices, $x_k$ is the state vector, $y_k$ is the output vector and $u_k$ is the input vector. The state $x_k$ and output vector $y_k$ of the state space model include the system outputs, which are the state temperatures. The state matrix $A$ and input matrix $B$ together include the building parameters, matrix $C$ outputs the state values and matrix $D$ transfers the input values. Table 2.4 lists the variables that are included by the input vector $u_k$ and output vector $y_k$ for this research. As can be seen, heating and cooling power $P$ is an input of the building model, because it is a controlled heat gain that influences the state temperatures. Further, the lumped RC building model only considers the building envelope and indoor air. Hence, a floor system and control model was coupled to the building model, as described later in step d).

To calculate the irradiation on the window surface $I_{\text{rad}}$ from the global horizontal irradiation $I_{\text{glob}}$, geometrical information, time of the day and cloud cover data, use was made of the Perez direct/diffuse split model [33].

#### Table 2.3: Overview required inverse modeling processes for each lumped RC building model

<table>
<thead>
<tr>
<th>Model development</th>
<th>Model validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step c) and d)</td>
<td>Step e)</td>
</tr>
<tr>
<td>(NEN 5050 E climate)</td>
<td>(NEN 5050 ref TO5 climate)</td>
</tr>
<tr>
<td>Case variant 1</td>
<td>Case variant 1</td>
</tr>
<tr>
<td>Case variant 4</td>
<td>Case variant 4</td>
</tr>
<tr>
<td>Case variant 5</td>
<td>Case variant 5</td>
</tr>
<tr>
<td>Case variant 2</td>
<td>Case variant 2</td>
</tr>
<tr>
<td>Case variant 3</td>
<td>Case variant 3</td>
</tr>
</tbody>
</table>

#### Table 2.4: Inputs and outputs of the lumped RC building model

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inputs ($u_k$)</td>
<td></td>
</tr>
<tr>
<td>$T_e$</td>
<td>Ambient temperature [$^\circ\text{C}$]</td>
</tr>
<tr>
<td>$I_{\text{rad}}$</td>
<td>Irradiation on window plane [$\text{W}\cdot\text{m}^{-2}$]</td>
</tr>
<tr>
<td>$Q_{\text{int}}$</td>
<td>Internal heat gains [W] (lighting, equipment, people)</td>
</tr>
<tr>
<td>Outputs ($y_k$)</td>
<td></td>
</tr>
<tr>
<td>$P$</td>
<td>Heating and cooling power [W]</td>
</tr>
<tr>
<td>$T_f$</td>
<td>Indoor air temperature [$^\circ\text{C}$]</td>
</tr>
<tr>
<td>$T_{\text{eff}}$</td>
<td>Effective floor temperature [$^\circ\text{C}$]</td>
</tr>
</tbody>
</table>
Selection of the RC building network configuration

To simulate the thermal behavior of a medium- or heavy-weight building exposed to environmental disturbances, an RC network with at least 2 resistances and 2 capacitances (R2C2) is probably required, to consider the effect of the building envelope capacity on the indoor air temperature [18, 28, 32, 34]. For buildings equipped with a thermal active floor, an additional floor capacitance is required. Subsequently, the application of the Perez direct/diffuse split model also requires an extra parameter $\text{win}$. Therefore, the lowest resolution RC building model configuration, considered in this research, consisted of 7 parameters ($j = 7$): 3 capacitances $C$, 3 effective conductances $G$ (where $G = R^{-1}$) and 1 parameter describing the effective window area $\text{win}$. The G3C3A1 network, as shown in black in figure 2.4, models the influence of the five considered physical phenomena on the indoor air temperature as follows:

- $T_e$ influences $T_i$ by the heat exchange through $1/G_{\text{ext}}$ (both transmission and wind-dependent infiltration are considered)
- The solar irradiation on the west façade $\text{Irrad}$ reaches an effective window area $A_{\text{win}}$, whereafter it heats up $T_{\text{w}}$
- The internal gains (lighting, equipment and people) are directly released to $T_i$
- The heating and cooling power $P$ is delivered to the floor state $T_f$. It then influences $T_i$ by the heat exchange trough $1/G_{\text{fl}}$
- The envelope capacitance $C_{\text{env}}$ considers the influence of the building’s thermal mass on $T_i$

Extra parameters were added to this R3C3A1 model to model the effect of the physical phenomena on the indoor air temperature in more detail. The parameters tested in this research are shown in grey in figure 2.4. An extra state $C_{\text{ot}}$ was added to distinguish radiative and convective solar gains (a). An extra state $C_{\text{ot}}$ to consider the temperature gradient in the thermal active floor (b). Both extra state $C_{\text{ot}}$ and resistances ($1/G_{\text{ot}}$ & $1/G_{\text{ot}}$) to differentiate transmission through the building envelope and infiltration (c). Appendix IV lists these lumped RC building networks and the association set of ODEs and state-space matrices.

2.3 Inverse modeling process

The objective of the inverse modeling process was to find the combination of resistance, capacitance and window area parameters $\chi$ that minimizes the root-mean-squared error (RMSE$_h$) over 8760 data values with a time interval of 3600 s, between the lumped RC model’s estimated ($\bar{T}_i$) and emulated ($T_i$) indoor air temperature. Eq. (3) defines this objective function $J$ [16]:

$$J(\chi) = \frac{1}{n} \sum_{k=1}^{n=8760} (\bar{T}_{ik} - T_{ik})^2$$

The search space of the parameter values is constrained to be one magnitude higher/lower than their approximated physical properties [29]. Table 2.5 shows an overview of the upper and lower parameter bounds for the G3C3A1 model. Parameter bounds of the extra parameters presented in grey in figure 2.4 can be found in appendix IV.

### Table 2.5: Lower and upper bounds G3C3A1 model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Lower bound</th>
<th>Upper bound</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_{\text{ext}}$</td>
<td>10</td>
<td>1000</td>
<td>[W·K$^{-1}$]</td>
</tr>
<tr>
<td>$G_{\text{int}}$</td>
<td>6</td>
<td>600</td>
<td>[W·K$^{-1}$]</td>
</tr>
<tr>
<td>$G_{\text{fl}}$</td>
<td>30</td>
<td>3000</td>
<td>[W·K$^{-1}$]</td>
</tr>
<tr>
<td>$C_{\text{ext}}$</td>
<td>1.00</td>
<td>100</td>
<td>[MJ·K$^{-1}$]</td>
</tr>
<tr>
<td>$C_{\text{int}}$</td>
<td>0.02</td>
<td>2</td>
<td>[MJ·K$^{-1}$]</td>
</tr>
<tr>
<td>$C_{\text{fl}}$</td>
<td>0.6</td>
<td>60</td>
<td>[MJ·K$^{-1}$]</td>
</tr>
<tr>
<td>$A_{\text{win}}$</td>
<td>0.5</td>
<td>50</td>
<td>[m$^2$]</td>
</tr>
</tbody>
</table>

For the identification of all lumped RC networks, the three-stage inverse modeling approach presented by Kramer et al. (2013) [28] was used. As illustrated in figure 2.5, the upper and lower bounds of each parameter were set preliminary. Subsequently, three algorithms of the Matlab global optimization toolbox [35] were deployed. Finally, the inverse modeling solution $\chi$ was checked. A new inverse modeling process was executed, if one or more parameter values of this solution coincided with either the upper or lower parameter bound.

![Figure 2.4: Initial RC model configuration (black) and extra parameters (grey)](image1)

![Figure 2.5: Three-stage inverse modeling approach for the identification of the lumped RC building model parameters [28]](image2)
refinement was performed by the Matlab direct search optimizer PatternSearch. Thirdly, the Matlab solver Fmincon was used to check the solution. Details about the algorithms can be found in ref. [35]. After the third stage, the bounds were checked to ensure that the identified parameter values did not coincide with these bounds, which might indicate that the search space was too small. Algorithm settings recommended by ref. [28, 35–37] were used for this research. Table 2.6 gives an overview of the main GA settings. Appendix V lists the algorithm settings of all three algorithms.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Population size</td>
<td>15 * number of parameters ( j )</td>
</tr>
<tr>
<td>Number of elite individuals</td>
<td>0.05 * population size</td>
</tr>
<tr>
<td>Crossover fraction</td>
<td>0.8</td>
</tr>
<tr>
<td>Stopping criteria</td>
<td>&lt; 0.001 % decrease in last 50 generations, OR max. generations: 30 * number of parameters ( j )</td>
</tr>
</tbody>
</table>

The inverse modeling process utilized the ambient temperature \( T_a \) of the NEN 5060E climate dataset, the calculated solar irradiation on the window \( \text{Irrad} \) (Perez model), the predefined internal heat gains \( Q_{\text{in}} \) and the emulated heating and cooling power \( P \) as inputs. The emulated indoor air temperature \( T_i \) was set as output of the lumped RC building model. The RC network and parameter space were predefined as illustrated in figure 2.6.

\[
u_k(t) = \begin{bmatrix} T_e \\ \text{Irrad} \\ Q_{\text{in}} \\ P \end{bmatrix}
\rightarrow \text{RC network with bounds: } x_1 \times 10^{-3} < x_j < x_2 \times 10^{-3}
\rightarrow y_k(t) = \begin{bmatrix} T_i \end{bmatrix}
\]

Figure 2.6: Inputs, system and output of lumped RC building model for identification of its parameters

When comparing table 2.3 to figure 2.6, it is notable that only one output variable was used in the inverse modeling process \( (T_i) \), while two output variables of the lumped RC model were used in this research for heating and cooling demand estimations \( (T_i \) and \( T_f) \). However the floor state temperature in the RC model is an 'effective' temperature, which cannot be verified with measured (or emulated) data, Yu et al. (2010) [16] showed that it is possible to use this effective temperature for the modeling of the floor system model.

### 2.4 Coupled simulation model

For the low-resolution model, the control strategy and the floor system, as applied in the virtual case study building were modeled. For this floor system model, an additional parameter was identified, before the lumped RC building model was coupled to this floor system model and control block. Finally, the coupled model was used to estimate indoor air temperatures and heating and cooling demands.

Identification effective heat resistance floor system

The floor system model estimated return water temperatures by using eq. (4), according to Yu et al. (2010) [16].

\[
T_{w,\text{ret}}(t) = T_{i}(t) + (T_{w,\text{sup}}(t) - T_{i}(t)) e^{-\frac{g_{\text{pipes}}}{m_{\text{w}}}}
\]

In this equation, the parameter \( g_{\text{pipes}} \) (effective heat resistance between the water in the floor system and \( T_f) \) was identified preliminary using the data shown in figure 2.7, according to eq. (5).

\[
g_{\text{pipes}} = \frac{1}{n} \sum_{k=1}^{n} \left( -\bar{m}_{\text{w}} e_{\text{w}} \ln \frac{T_{w,\text{ret},k}}{T_{w,\text{sup},k}} - \frac{T_{i,k}}{T_{f,k}} \right)
\]

Figure 2.7: Data used for identification of \( g_{\text{pipes}} \)

Compared to the inverse modeling process of section 2.3, the RC network parameters were known and only one parameter was identified in this process. First, the lumped RC building model with its identified parameters were used to estimate the effective floor temperature \( T_{f'} \). Subsequently, \( g_{\text{pipes}} \) was identified manually using equation (5).

The virtual case study building provides data with a time interval of 3600 s, where \( P(t) \) is the average power output over this interval and \( T_{w,\text{ret}}(t) \) is a spot measurement at each interval. Since time integrated \( (P) \) and periodic outputs \( (T_{w,\text{ret}}) \) of the case study building were both used in the steady-state equation, this equation is only valid when the system is activated throughout the entire time interval. Therefore, eq. (5) only considered time steps in which this occurred to the case study building.

Coupling of models

Figure 2.8 shows how the lumped RC building model was coupled to the control and floor system model. Compared to the diagrams of figure 2.6 and 2.7, the input \( P \) (from virtual case study building) is replaced by the estimated power \( \hat{P} \). Hence, an algebraic loop was formed among the building, floor system and control model. In this loop, the outputs of the lumped RC building model \( (\hat{T}_i \) and \( \hat{T}_f) \) at time step \( t \) are the
inputs of the control and floor system model respectively at time step \( t_2 \). The estimated power \( \hat{P} \) by the floor system model at time step \( t_2 \) was then used as input of the lumped RC building model at \( t_2 \).

Figure 2.8: Schematic diagram of coupled simulation model with its inputs, variables and outputs

Estimation indoor air temperatures & energy demands

The control determined whether power was required by comparing the estimated indoor air temperature \( \hat{T}_i \) (from lumped RC building model) with the heating and cooling set points \( T_{i,\text{set}} \). When \( \hat{T}_i \) applied to this setting, the supply water temperature \( T_{w,sup} \) was determined from the heating curve (as function of \( T_e \)). \( T_{w,sup} \) was then compared with \( \hat{T}_i \) to determine whether \( T_{w,sup} \) is higher than \( \hat{T}_i \) in case of heating and vice versa for cooling. If \( \hat{T}_i \) applies to this setting as well, the floor system model is activated and \( T_{w,sup} \) is set for the floor system model. If not, the estimated power equals zero.

If the floor system was activated by the control, the floor system model determined the return water temperature as described by eq. (6), which is an adjusted equation of eq. (4) [16]. Subsequently, the estimated power was determined using eq. (7).

\[
\hat{P}(t) = m_{w} c_{w} (\hat{T}_{w,sup}(t) - \hat{T}_{w,rel}(t))
\]

To reduce the simulation error due to the difference in time step, the simulation was executed with a time interval of 60 s. Consequently, the input data (time interval of 3600 s) was interpolated and the indoor air temperatures were obtained each hour (every 60 simulations equals one hour). Furthermore, the estimated power was post-processed: heating and cooling power was obtained by splitting this power in a positive (\( P_{\text{heat}} \)) and negative part (\( P_{\text{cool}} \)). Energy demands were then determined by integrating these two parts over the time interval \( t \), according to eq. (8) and (9).

\[
E_{\text{heat}}(t) = \int_{0}^{t} P_{\text{heat}}(t) \, dt \tag{8}
\]

\[
E_{\text{cool}}(t) = \int_{0}^{t} P_{\text{cool}}(t) \, dt \tag{9}
\]

2.5 Model validation

The model validation consisted of two tests. A cross-correlation test was executed to evaluate the model’s ability to describe the influence of each separate model input on the estimated indoor air temperature [38]. And a comparison test based on a set of performance metrics was performed to assess the model’s ability to estimate heating and cooling demands.

Cross-correlation test

The residual of the simulation model is the part of the output signal not explained by the model, as defined by equation (10):

\[
\epsilon = T_i - \hat{T}_i \tag{10}
\]

The cross-correlation test indicates whether \( \epsilon \) is statistically independent from each separate input. The cross-correlation coefficient \( r \) describes the correlation between two time series as function of lag (time-offset), according to eq. (11) [39]. For time series length \( n = 8760 \) and lag \( t = \{-24 : 24\} \):

\[
r(t) = \frac{1}{n} \sum_{i=1}^{n-1} (u_i - \bar{u})(\epsilon_{i+t} - \bar{\epsilon}) \sqrt{\frac{1}{n} \sum_{i=1}^{n} (u_i - \bar{u})^2 \cdot \frac{1}{n} \sum_{i=1}^{n} (\epsilon_i - \bar{\epsilon})^2}
\]

This test was executed before the RC building model was coupled to the floor system and control model, so no feedback is obtained between input \( u(t) \) and past residuals \( \epsilon \) [40]. Therefore, climate data and virtual case study building data were used as inputs for this test as shown in figure 2.9.
Cross-correlation settings were retrieved from studies in which this test was deployed for the validation of RC building models [28, 41, 42]. The coefficients were tested on a 95% confidence interval for 24 positive and negative lags (+1 day and -1 day shift resp.) using the Matlab system identification toolbox [43]. Ljung (1987) [44] states that a significant cross-correlation coefficient for a specific input indicates that the influence of this input on the indoor air temperature is incorrectly described by the model.

Comparison test based on performance metrics
The model’s performance to estimate heating and cooling demand was assessed based on the percentage error (PE) between the estimated (E_{heat,a}) and emulated annual heating demand (E_{heat,a}) according to eq. (12) [45, 46]. A similar equation was used for the calculation of the percentage error of the estimated annual cooling demand (13). The model confirmed for this research if both (12) and (13) are less than 10% and all cross-correlation coefficients are within the 95% confidence interval or coinciding one of the two interval lines.

\[
P_{E_{heat,a}} = 100 \cdot \frac{E_{heat,a} - \hat{E}_{heat,a}}{E_{heat,a}}
\]  \hspace{1cm} (12)

\[
P_{E_{cool,a}} = 100 \cdot \frac{E_{cool,a} - \hat{E}_{cool,a}}{E_{cool,a}}
\]  \hspace{1cm} (13)

3. RESULTS
This chapter firstly displays (a) the virtual case study building’s data that was applied in the remaining methodology steps. Second, (b) the lumped RC network that was found to suffice for this research is presented. Subsequently, (c) the inverse modeling process results and (d) the coupled model’s estimation results are shown for the lumped RC network of step b. Finally, (e) the results of the validation method are listed.

3.1 Virtual case study building data
Figure 3.1. displays the emulated data that was used for the inverse modeling process of the lumped RC building models. The emulated annual heating and cooling demands are used for the model validation are shown in figure 3.2.

Figure 3.2: Emulated annual heating and cooling demand of the five case study building variants for the validation dataset (NEN 5060 ref TO5)

Figure 3.1: Emulated indoor air temperature and heating and cooling power of case variants 1-3 for a winter (left) and summer (right) week of the identification dataset (NEN 5060 E)
3.2 Lumped RC building model

Table 3.1 lists the objective function values determined for the investigated lumped RC building networks. After coupling these identified building models to the floor system model and control, it was found that the G4C4A1 network was appropriate to confidently estimate indoor air temperatures as well as annual heating and cooling demands. The motivation for this selection is described in section 3.5 and a complete overview of the results can be found in appendix VI. The resistance-capacitance network of this G4C4A1 network is shown in figure 3.3.

<table>
<thead>
<tr>
<th>Variant</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>G3C3A1 network (fig. 2.4)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>initial model</td>
<td>0.518</td>
<td>0.621</td>
<td>0.657</td>
</tr>
<tr>
<td>G4C4A1 network (fig. 3.3)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>initial model + A&lt;sub&gt;win&lt;/sub&gt; + a</td>
<td>0.515</td>
<td>0.619</td>
<td>0.655</td>
</tr>
<tr>
<td>G4C4A2 network (fig. 3.4)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>initial model + a, b, c</td>
<td>0.492</td>
<td>0.601</td>
<td>0.642</td>
</tr>
<tr>
<td>G7C6A2 network</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>initial model + a, b, c</td>
<td>0.492</td>
<td>0.601</td>
<td>0.642</td>
</tr>
</tbody>
</table>

The ODEs of the selected G4C4A1 network are shown below. The state-space matrices of this model can be found in the appendix at the end of this paper.

\[
\frac{dT}{dt} = \frac{A_{env}}{C_{env}} (T_i - T_{env})
\]

\[
\frac{d\hat{T}_i}{dt} = \frac{g_{fast}}{C_i} (T_e - \hat{T}_i) + \frac{g_{env,i}}{C_i} (\hat{T}_{env} - \hat{T}_i) + \frac{g_{int}}{C_i} \cdot (\hat{T}_{int} - \hat{T}_i) + \frac{g_{fl}}{C_i} (\hat{T}_{fl} - \hat{T}_i) + \frac{1}{C_i} q_{int}
\]

\[
\frac{d\hat{T}_{int}}{dt} = \frac{g_{int}}{C_{int}} (\hat{T}_i - \hat{T}_{int}) + \frac{A_{win}}{C_{int}} \frac{Q_{irrad}}{A_{win}}
\]

\[
\frac{d\hat{T}_{fl}}{dt} = \frac{g_{fl}}{C_{fl}} (\hat{T}_i - \hat{T}_{fl}) + \frac{1}{C_{fl}} q
\]

The objective function values obtained for this network were approximately 0.002 °C lower than for the lowest resolution G3C3A1 network. The effect of this difference on the model’s level of confidence was found to be significant, as is described in section 3.5. The lowest objective function values were found for the G4C4A2 network, which is shown in figure 3.4. Either the addition of an extra state \(C_{22}\) (b) or both extra states \(C_{env2}\) and resistances \(1/G_{env2}\) & \(1/G_{env3}\) (c) to this network did not further enhance the objective function value. Moreover, the addition of both (b) and (c) did not cause improvements.

3.3 Inverse modeling process

Table 3.2 lists the identified RC network parameter values for the G4C4A1 model. The parameter values that are different for one variant compared to the other two variants are highlighted in grey. For virtual case study building variant 2, relatively higher values were identified for parameters \(G_{env}\) and \(C_{env}\). This can be explained by the higher infiltration rate for this variant. The higher values for parameters \(G_{int}\) and \(A_{win}\) of case variant 3 are probably related to the higher g-value.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Variant 1</th>
<th>Variant 2</th>
<th>Variant 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>(G_{env})</td>
<td>36.1</td>
<td>52.3</td>
<td>48.3</td>
</tr>
<tr>
<td>(G_{int})</td>
<td>62.0</td>
<td>73.2</td>
<td>62.0</td>
</tr>
<tr>
<td>(G_{int})</td>
<td>70.1</td>
<td>20.1</td>
<td>32.7</td>
</tr>
<tr>
<td>(G_{int})</td>
<td>640.6</td>
<td>637.2</td>
<td>648.2</td>
</tr>
<tr>
<td>(C_{env})</td>
<td>12.6</td>
<td>24.0</td>
<td>15.3</td>
</tr>
<tr>
<td>(C_{int})</td>
<td>0.2</td>
<td>0.2</td>
<td>0.1</td>
</tr>
<tr>
<td>(C_{int})</td>
<td>0.1</td>
<td>5.6</td>
<td>5.6</td>
</tr>
<tr>
<td>(C_{fl})</td>
<td>4.5</td>
<td>4.6</td>
<td>7.2</td>
</tr>
</tbody>
</table>

3.4 Coupled simulation model

Figure 3.5 displays the G4C4A1 model’s ability to estimate the dynamic behavior of the indoor air temperatures for the first case study building variant. In addition, figure 3.6 compares the estimated and emulated daily, weekly and monthly heating and cooling demand estimated for the first case study building variant.

Table 3.3 quantifies the model’s estimation accuracy by the RMSE between the estimated and emulated indoor air temperature. Furthermore, the Normalized Root Mean Squared Error (NRMSE) between the estimated and emulated power over 8760...
hourly data samples is shown. Braun and Chaturvedi (2002) [13] defined this performance metric as:

\[
NRMSE_{P,h} = 100 \cdot \frac{1}{N} \sum_{k=1}^{N} (P_k - \bar{P}_k)^2
\]

\[
\bar{P}_k = \frac{1}{N} \sum_{k=1}^{N} P_k
\]

(14)

Table 3.3: Performance metrics of coupled simulation model with G4C4A1 network (validation data)

<table>
<thead>
<tr>
<th>Variant</th>
<th>RMSE(_{T,A}) [°C]</th>
<th>NRMSE(_{P,h}) [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variant 1</td>
<td>0.33</td>
<td>4.61</td>
</tr>
<tr>
<td>Variant 2</td>
<td>0.42</td>
<td>7.70</td>
</tr>
<tr>
<td>Variant 3</td>
<td>0.43</td>
<td>6.36</td>
</tr>
<tr>
<td>Variant 4</td>
<td>0.37</td>
<td>4.59</td>
</tr>
<tr>
<td>Variant 5</td>
<td>0.29</td>
<td>4.85</td>
</tr>
</tbody>
</table>

3.5 Model validation

This section first considers the cross-correlation test. Additionally, the results of the comparison test based on the set of performance metrics are presented.

Cross-correlation test

Figure 3.7 shows an overview of the cross-correlation coefficients found for the G4C4A1 network. The gray area represents the tolerated bandwidth (95% confidence interval). For all 5 case variants, the coefficients of all inputs are either within the tolerated bandwidth or approximately coinciding with the lower confidence interval line. This indicates that the indoor air temperatures are estimated correctly. Therefore, the estimated heating and cooling demands were considered as confident. The restrictions for the deployment of the cross-correlation test for this research (due to the use of emulated data) are further discussed in 4.4.
It was found that for case variants 1, 4 and 5, a decline in the RC building model resolution causes cross-correlation coefficients up to two times the tolerated bandwidth.

Validation coupled simulation model

Figure 3.9 demonstrates the method’s ability to estimate annual heating and cooling demands for the five case study building variants. Table 3.4 lists the percentage errors obtained.

<table>
<thead>
<tr>
<th>Variant</th>
<th>PE\textsubscript{heat} [%]</th>
<th>PE\textsubscript{cool} [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variant 1</td>
<td>-1.25</td>
<td>-2.94</td>
</tr>
<tr>
<td>Variant 2</td>
<td>-2.38</td>
<td>-3.83</td>
</tr>
<tr>
<td>Variant 3</td>
<td>-4.13</td>
<td>-0.65</td>
</tr>
<tr>
<td>Variant 4</td>
<td>+1.82</td>
<td>-2.95</td>
</tr>
<tr>
<td>Variant 5</td>
<td>-1.34</td>
<td>-2.65</td>
</tr>
</tbody>
</table>
As can be seen, the maximum percentage deviation obtained is considerably lower than the adopted 10%. It was found that this criterion was already confirmed with the lowest resolution RC model proposed (G3C3A1 network).

Figure 3.9: Estimated and emulated annual heating and cooling demands for validation dataset

4. DISCUSSION
A methodology consisting of 5 main steps was implemented to develop a simulation model for a reliable estimation of the indoor air temperature and the heating and cooling demand. Each step is discussed successively in this chapter.

4.1 Virtual case study building
A typical Dutch office building with a floor heating and cooling system was defined and the uncertain building parameters of this building were selected and quantified. Based on this building and these parameters, five case variants were emulated using the high-resolution building simulation tool Vabi Elements.

Despite the case study building was a representation of a typical Dutch office building, the complexity of the configuration was considerably lower compared to that of a Dutch office building in practice. Firstly, the case study building set the ventilation rate to zero, while office buildings generally consist of a mechanical ventilation system and options for natural ventilation. Since this ventilation accounts for a significant share in the heat balance of the building’s indoor air, it should be taken into account when defining a lumped RC building network (step b of the methodology) for a typical office building [22]. Secondly, the case study building included one room, while existing office buildings generally consists of multiple rooms and zones. For the case study building of this research, the proposed methodology considered one uniform indoor air temperature for the development of the lumped RC building model. It was shown that this is also possible for a test laboratory and a dwelling consisting of three rooms heated by a radiator system [21]. However, further research is required to determine whether this is also possible for multi-zone office buildings with floor heating and cooling systems.

The four uncertain building parameters that were considered in this research only represent a few of the numerous uncertain building parameters in an existing office building. Consequently, the methodology is assumed valid for estimation of expected energy distributions due to these four parameters, but invalid for energy distributions estimations due to other uncertain parameters (e.g. occupancy and system time schedules [19]).

Even though a high-resolution tool was utilized to emulate the case study building variants, the level of confidence of the developed model in this research will presumably change when it is deployed for existing buildings. Since the case study building was an emulated version of an existing building, among others all input variables (e.g. internal heat gains) were known, surface and air temperatures were assumed uniformly distributed and the floor heating and cooling system had a uniform water supply over the floor area. However, the methodology does not require adjustments for the application of the methodology.

4.2 Lumped RC building model
Based on the case study building configuration, a lumped RC building model configuration was defined that described the five physical phenomena shown in figure 2.3. A linear building model that consisted of only time-invariant parameters was used to model these phenomena. The model utilized lumped capacitances and resistances, which reduced the amount of model parameters. Starting with the lowest model resolution, the resolution was gradually increased.

The results demonstrated that this linear lumped RC building model, that consisted of only time-invariant parameters, could correctly model the influence of these phenomena on the dynamic behavior of the building’s indoor air temperature. The $\text{RMSE}_{\text{E}_{\text{heat}}}$ and $\text{PE}_{\text{heat}}$ increased by 0.1 °C and 1 % respectively as the wind-dependent infiltration rate doubled and the solar gains increased with 73 % (table 3.3 and 3.4). Probably due to the relatively high heat flow by infiltration, it was possible to consider the infiltration and transmission as a joint heat exchange between $T_e$ and $T_i$. The results prove that the Perez direct/diffuse split model was successfully deployed for the estimation of solar gains.

The simulation error remained the same when the internal heat gains or the temperature set points changed. This indicated that the radiative and convective internal heat gains could be modeled as one gain and the concrete floor as one capacitance.
4.3 Inverse modeling process
The three-stage inverse modeling approach presented by Kramer et al. (2013) [28] was used to identify the RC network parameter values. While the search space of each parameter set manually, the initial parameter values were defined by the GA. Stopping criteria were used for all three algorithms to enable acceptable calculation demands.

To test the reproducibility of the inverse modeling process, 100 identical inverse modeling process runs were executed. As shown in figure 4.1, all inverse modeling process resulted in an objective function value between 0.515 and 0.516 °C. It was found that all processes stopped due to the maximum amount of generation that was set as stopping criteria.

As shown by figure 2.1, the inverse modeling process could not find the solution with the lowest objective function value each inverse modeling process run; two solutions with a deviation of 0.001 °C compared to each other exist for the case variant. Table 2.1 list the identified RC network parameter values for these two solutions. The parameters that have an identified value for \( J = 0.516 \) °C with a factor of at least 1.5 higher compared to the value for \( J = 0.515 \) °C are displayed in grey. It was difficult to determine the cause of the difference in identified values, since use was made of effective parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>( J = 0.515 ) [°C]</th>
<th>( J = 0.516 ) [°C]</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \dot{G}_{\text{env}} )</td>
<td>36.1</td>
<td>480.4</td>
<td>[W·K(^{-1})]</td>
</tr>
<tr>
<td>( \dot{G}_{\text{fast}} )</td>
<td>62.0</td>
<td>62.0</td>
<td>[W·K(^{-1})]</td>
</tr>
<tr>
<td>( \dot{G}_{\text{int}} )</td>
<td>20.1</td>
<td>52.0</td>
<td>[W·K(^{-1})]</td>
</tr>
<tr>
<td>( \dot{G}_{\text{fl}} )</td>
<td>640.6</td>
<td>1219.0</td>
<td>[W·K(^{-1})]</td>
</tr>
<tr>
<td>( C_{\text{env}} )</td>
<td>12.6</td>
<td>13.8</td>
<td>[MJ·K(^{-1})]</td>
</tr>
<tr>
<td>( C_{\text{fast}} )</td>
<td>0.2</td>
<td>0.2</td>
<td>[MJ·K(^{-1})]</td>
</tr>
<tr>
<td>( C_{\text{int}} )</td>
<td>0.1</td>
<td>0.1</td>
<td>[MJ·K(^{-1})]</td>
</tr>
<tr>
<td>( C_{\text{fl}} )</td>
<td>5.6</td>
<td>5.6</td>
<td>[MJ·K(^{-1})]</td>
</tr>
<tr>
<td>( A_{\text{win}} )</td>
<td>4.5</td>
<td>4.5</td>
<td>[m(^2)]</td>
</tr>
</tbody>
</table>

The fact that these two solutions result in a different combination of building parameter is not a problem for a confident estimation of heating and cooling demands. However, these parameters values need to be identified correctly, since these values are of importance for the application of the method in future research steps: for the estimation of expected energy distributions, these parameters need to be identified correctly, since variations will be made in the parameter values to estimate these distributions.

The effect of this deviation in objective function value on the level of confidence is indicated by figure 4.2. The figure shows the cross-correlation coefficients found for the lumped RC network identified with objective function value 0.516 °C.

When comparing the cross-correlation coefficients in figure 3.7 for variant 1 with the coefficients in figure 4.2, it can be concluded that the uncertainty in the inverse modeling process significantly influences the model’s level of confidence.

The majority of the inverse modeling processes in the consulted literature did not have a deviation in its solution, because initial parameter values were set manually [13, 16, 21, 32, 47]. Studies in which the initial parameter values set by an algorithm did not mention it [22, 29, 48]. Kramer et al. (2013) [28] discussed the distribution in the inverse modeling process solution and found a dispersion of 0.05 °C in the objective function values for 100 identical inverse modeling processes (RMSE = 0.67 - 0.72 °C). Even though this study considers a different case study building configuration, this indicates that the inverse modeling process in Kramer’s study did not show reproducible results either.

Since the reproducibility of the inverse modeling process is relevant for future research steps and this methodology step is rarely discussed or found problematic in literature, additional research for this step might be required.

4.4 Coupled simulation model
For the determination of the heating and cooling power, the control strategy and the floor system were modeled, as applied in the virtual case study building. Therefore, it was necessary to model the floor system (water pipes + water) separately from
the concrete floor. Firstly, an additional parameter was identified for this floor system model. Secondly, the lumped RC building model was coupled to this floor system model and control block and the coupled model was then used to estimate indoor air temperatures and heating and cooling demands.

For the determination of the floor system model parameter $g_{pipes}$, the time integrated variable $P_t$, the spot measurement variable $T_{reset}$ and the effective floor temperature $T_{eff}$ were used in a steady-state equation. To indicate the uncertainty due to the use of this equation, figure 4.3 shows the parameter values obtained for $g_{pipes}$ for the 100 identified G4C4A1 networks, according to eq. (5).

![Figure 4.3: Parameter values found for $g_{pipes}$ for the 100 identified G4C4A1 networks for case variant 1](Image)

As can be seen, despite the fact that a time integrated variable, a spot measurement variable and an effective variable were used in eq. (5), the variation in the parameter value is limited (2 %). Therefore, it can be concluded that this equation is reliable and that $g_{pipes}$ can be used for the estimation of heating and cooling demands.

Since the floor system model is modeled separately from the concrete floor, two issues arose: the lumped RC building model needed to confidently estimate an ‘effective’ floor temperature; and a difference in time step existed between this estimated effective floor temperature and the floor system model’s output power. Figures 3.7 and 3.9 demonstrated that the methodology could correctly estimate indoor air temperatures and heating and cooling demands by using this effective temperature and a calculation time interval of 60 s.

### 4.5 Model validation

The model validation consisted of a residual analysis and a comparison test based on a set of performance metrics.

#### Residual analysis

The residual analysis in this research was applied on emulated data of the case study building. Due to the absence of noise in this data, the residual analysis excluded the autocorrelation test and only included the cross-correlation test [38]. Therefore, the residual analysis in this research indicated whether the influence of each separate input on the estimated indoor air temperature is correctly described, but did not indicate whether an adequate number of inputs was used. The absence of the autocorrelation test in this research was assumed negligible, because the $RMSE_{\text{th}}$ values in table 3.3 and the plots in figure 3.5 indicated that no inputs were missing to estimate the dynamic behavior of the indoor air temperature. However, the addition of the autocorrelation test to the methodology, when it is applied on real measured data, might enhance the method’s justification.

#### Comparison test based on performance metrics

The performance metrics $PE_{\text{heat,a}}$ and $PE_{\text{cool,a}}$ were used to assess the ability of the coupled simulation model to estimate heating and cooling demands. These performance metrics gave a clear indication about the model’s ability to estimate these demands on long-time intervals (seasonal, annual). However, it does not indicate the performance of the model for small-time scales. As illustrated by table 4.2 for variant 2, the use of these metrics caused difficulties to evaluate the investigated coupled simulation models. The two highest resolution models estimated heating and cooling power with a $NRMSEP_{i,h}$ which is circa 0.8 % lower compared to the two lowest resolution models. In contrary, the $PE_{\text{heat,a}}$ and $PE_{\text{cool,a}}$ were 1 and 3 % higher respectively. For the estimation of annual heating and cooling demands, these $PE_{\text{heat,a}}$ and $PE_{\text{cool,a}}$ can be used. However, for estimations of heating and cooling demands on small-time scales (hours, days), the $NRMSEP_{i,h}$ might be a more appropriate performance metric for the model validation.

<table>
<thead>
<tr>
<th align="center">Table 4.2: Performance metrics found for investigated RC networks for variant 2</th>
</tr>
</thead>
<tbody>
<tr>
<td align="center">$J$ [$^\circ\text{C}$]</td>
</tr>
<tr>
<td align="center">initial model</td>
</tr>
<tr>
<td align="center">$G4C4A1$ network</td>
</tr>
<tr>
<td align="center">initial model $+ a$</td>
</tr>
<tr>
<td align="center">$G6C7A2$ network</td>
</tr>
</tbody>
</table>

### 5. CONCLUSIONS

To confidently estimate hourly indoor air temperatures and annual heating and cooling demands of a single-zone office building with a floor heating and cooling system, a lumped RC building model with at least 9 parameters is required: a model consisting of four capacitances, four resistances and one parameter describing the total window area. A model consisting of 7 parameters can already estimate annual heating and cooling demands with a
percentage error less than 5%. However, this model does not confidently estimate the influence of the ambient temperature and heating and cooling power on the indoor air temperature. Since indoor air temperatures strongly affect the heating and cooling demands, a correct estimation of these temperatures is required to perform a confident estimation of the annual heating and cooling demands. Enhancing the model resolution with one extra parameter to define the window area increases the level of confidence, while increasing the number of capacitances and resistances does not enhance this level.

Recommendations

This research was part four-stage approach procedure proposed by Henze et al. (2015) to apply a lumped RC network in combination with inverse modeling for operational energy use evaluations [19]. Further research is required for this first stage to test the method’s ability to consider the complex heat balance of existing office buildings. It is recommended to increase the case study building’s complexity gradually in future research steps to attain clear insight in the effect of each influencing factor (affecting the indoor air temperature’s heat balance) on the method’s level of confidence. At first, mechanical ventilation should be included in the single-zone case study building. Secondly, the case study building can be split up in multiple rooms or zones. Thirdly, the methodology can be applied on measured data of multi-zone office buildings.

For this third step, the autocorrelation test can be added to the residual analysis. When the methodology is applied for the estimation of hourly or daily heating and cooling demands, the NRMSE_{R,S} might be an appropriate performance metric for the model validation. For the application of the methodology for the estimation of expected energy distributions, further research is required to enable a reproducible combination of identified network parameters.

ACKNOWLEDGEMENT

This research has emerged from a collaboration between the Eindhoven University of Technology and DWA. The DWA contribution was supported by the Stichting Innovatie Alliantie (SIA) through the project Installaties 2020. The author wish to thank all supervisors for their advice during the project.

APPENDICES

I. NEN 5060 climate data
II. Case study building description
III. Emulated data (Vabi results)
IV. Lumped RC building networks
V. System identification settings
VI. Results different RC model configurations
VII. Matlab and Simulink model scripts

The state space matrices of the G4C4A1 network are:

\[
A = \begin{bmatrix}
-g_{env} & g_{env} & 0 & 0 \\
g_{env} & -g_{int} & -g_{int} & -g_{fl} \\
0 & g_{fl} & -g_{int} & 0 \\
0 & 0 & 0 & -g_{fl}
\end{bmatrix}
\]

\[
B = \begin{bmatrix}
0 & 0 & 0 & 0 \\
g_{env} & 0 & \frac{1}{c_{fl}} & 0 \\
0 & A_{gin2} & 0 & 0 \\
0 & 0 & 0 & \frac{1}{c_{fl}}
\end{bmatrix}
\]

\[
C = \begin{bmatrix}
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0
\end{bmatrix}
\]

\[D = 0\]

With state matrix \(\hat{x}_k\):

\[\hat{x}_k = [T_{env} \ T_i \ T_{int} \ T_{fl}]\]

And input matrix \(u_k\):

\[u_k = [T_e \ \text{irrad} \ Q_{int} \ P]\]

REFERENCES

APPENDIX I: NEN 5060 CLIMATE DATA

Winter week: February 11 – 17
Summer week: August 5 – 12

NEN 5060 E (A2 for energy calculations)

- Ambient air temperature [°C]

- Cloud cover [-]

- Diffuse solar irradiation on the horizontal [W·m⁻²]

- Direct normal solar irradiation on the horizontal [W·m⁻²]
NEN 5060 ref TO5 (B2: annual probability of exceedance 5 \%)

- Ambient air temperature [°C]

Figure I.5: Ambient air temperature for a winter (left) and summer week (right) of the NEN 5060 ref. TO5 dataset

- Cloud cover [-]

Figure I.6: Cloud cover for a winter (left) and summer week (right) of the NEN 5060 ref. TO5 dataset

- Diffuse solar irradiation on the horizontal [W·m⁻²]

Figure I.7: Diffuse solar irradiation on the horizontal for a winter (left) and summer week (right) of the NEN 5060 ref. TO5 dataset

- Direct normal solar irradiation on the horizontal [W·m⁻²]

Figure I.8: Direct normal solar irradiation on the horizontal for a winter (left) and summer week (right) of the NEN 5060 ref. TO5 dataset
APPENDIX II: CASE STUDY BUILDING DESCRIPTION

VARIANT 1: BASE CASE VARIANT

II.1 Project Data

Climate file
Climate file: NEN5060 ref energie
Start date: 01-01-1906
End date: 31-12-1906
Number of calculated days: 365
National days and holidays: national days and holidays not taken into account.

Starting conditions
Solar radiation ground reflection (from climate file): 0.2
Calculated with shading
- Shading from own building: Yes
- Shading from building parts: No
- Shading from recessed windows: No
- Shading from surrounding buildings: No
- Canopies: No
Openable windows: Present
Windows open when indoor temperature is:

System information
Central Air handling: Absent
Local air handling: Present
Mechanical ventilation: balanced

II.2 Room statistics

Room use: other
Room type: habitable room
Counting hours: 100% on
Summer clothing: 0.70 CLO
Winter clothing: 0.90 CLO
Metabolism: 1.20 MET

Building parts

<table>
<thead>
<tr>
<th>Surface no.</th>
<th>In surface</th>
<th>Type</th>
<th>Description</th>
<th>Constr. #</th>
<th>Reverse side</th>
<th>Hor. [°]</th>
<th>Vert. [°]</th>
<th>Area [m²]</th>
<th>Reveal [mm]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>wall</td>
<td>Wall 1</td>
<td>1 outside</td>
<td>180</td>
<td>S</td>
<td>90</td>
<td>16.20</td>
<td>16.20</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>wall</td>
<td>Wall 1</td>
<td>1 outside</td>
<td>270</td>
<td>W</td>
<td>90</td>
<td>9.60</td>
<td>9.60</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>wall</td>
<td>Wall 1</td>
<td>1 outside</td>
<td>0</td>
<td>N</td>
<td>90</td>
<td>16.20</td>
<td>16.20</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>wall</td>
<td>Wall 1</td>
<td>1 outside</td>
<td>90</td>
<td>E</td>
<td>90</td>
<td>21.60</td>
<td>21.60</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>wall</td>
<td>Roof 1</td>
<td>2 outside</td>
<td>180</td>
<td>S</td>
<td>0</td>
<td>48.00</td>
<td>48.00</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>wall</td>
<td>Floor 1</td>
<td>3 ground</td>
<td>180</td>
<td>S</td>
<td>180</td>
<td>48.00</td>
<td>48.00</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>window frame</td>
<td>window 2</td>
<td>4 outside</td>
<td>270</td>
<td>W</td>
<td>90</td>
<td>5.40</td>
<td>5.40</td>
</tr>
<tr>
<td>8</td>
<td>7</td>
<td>glass</td>
<td>window 2</td>
<td>5 outside</td>
<td>270</td>
<td>W</td>
<td>90</td>
<td>5.40</td>
<td>5.40</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>window frame</td>
<td>window 2</td>
<td>4 outside</td>
<td>270</td>
<td>W</td>
<td>90</td>
<td>5.40</td>
<td>5.40</td>
</tr>
<tr>
<td>10</td>
<td>9</td>
<td>glass</td>
<td>window 2</td>
<td>5 outside</td>
<td>270</td>
<td>W</td>
<td>90</td>
<td>5.40</td>
<td>5.40</td>
</tr>
</tbody>
</table>

Total surface area [m²]

<table>
<thead>
<tr>
<th>Surface no.</th>
<th>Description</th>
<th>Orientation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Hor. [°]</td>
<td>Vert. [°]</td>
</tr>
<tr>
<td>8 window</td>
<td>270</td>
<td>90</td>
</tr>
<tr>
<td>10 window</td>
<td>270</td>
<td>90</td>
</tr>
</tbody>
</table>

Solar shading

<table>
<thead>
<tr>
<th>Surface no.</th>
<th>Description</th>
<th>Type</th>
<th>Control type</th>
<th>Day</th>
<th>Night</th>
</tr>
</thead>
<tbody>
<tr>
<td>8 window</td>
<td>None</td>
<td>None</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10 window</td>
<td>None</td>
<td>None</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Appendix II / VII
II.3 Overview of all constructions used

### Opaque constructions

<table>
<thead>
<tr>
<th>Constr</th>
<th>Ref #</th>
<th>Description</th>
<th>Type</th>
<th>Input</th>
<th>Thickness</th>
<th>Rc</th>
<th>U</th>
<th>Mass</th>
<th>Thermally active</th>
<th>Material layers?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ON1_Wall</td>
<td>wall</td>
<td>Yes</td>
<td>232</td>
<td>3.34</td>
<td>0.29</td>
<td>146</td>
<td>No</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ON3_Roof</td>
<td>roof</td>
<td>Yes</td>
<td>154</td>
<td>3.32</td>
<td>0.28</td>
<td>21</td>
<td>No</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>OR12_Floor</td>
<td>floor</td>
<td>Yes</td>
<td>1080</td>
<td>25.07</td>
<td>0.03</td>
<td>122</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>ON24_window_kozijn</td>
<td>window</td>
<td>Yes</td>
<td>3.80</td>
<td>3.80</td>
<td>0.04</td>
<td>0.04</td>
<td>No</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Transparent constructions

<table>
<thead>
<tr>
<th>Constr</th>
<th>Ref #</th>
<th>Description</th>
<th>Position</th>
<th>Absorption</th>
<th>Emission</th>
<th>Convection</th>
<th>U</th>
<th>ZTA</th>
<th>CF</th>
<th>Sun Protection</th>
<th>Placing</th>
<th>Circuit</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>ON4_window_glas</td>
<td>1</td>
<td>0.374</td>
<td>0.90</td>
<td>23.00</td>
<td>1.10</td>
<td>0.04</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>0.026</td>
<td>2.70</td>
<td>1.10</td>
<td>0.04</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>2.70</td>
<td>3.0</td>
<td>1.10</td>
<td>0.04</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
<td>3.00</td>
<td>1.10</td>
<td>0.04</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Construction # 1, ON1_Wall

<table>
<thead>
<tr>
<th>Description</th>
<th>Thickness</th>
<th>Lambda</th>
<th>Specific mass</th>
<th>Specific Heat</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wood_Siding</td>
<td>0.009</td>
<td>0.140</td>
<td>530</td>
<td>900 Outside</td>
</tr>
<tr>
<td>Fiberglass_quilt</td>
<td>0.123</td>
<td>0.040</td>
<td>12</td>
<td>840</td>
</tr>
<tr>
<td>Concrete_block</td>
<td>0.100</td>
<td>0.510</td>
<td>1400</td>
<td>1000 Inside</td>
</tr>
</tbody>
</table>

#### Construction # 2, ON3_Roof

<table>
<thead>
<tr>
<th>Description</th>
<th>Thickness</th>
<th>Lambda</th>
<th>Specific mass</th>
<th>Specific Heat</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roof_Deck</td>
<td>0.019</td>
<td>0.140</td>
<td>530</td>
<td>900 Outside</td>
</tr>
<tr>
<td>Fiberglass_quilt</td>
<td>0.125</td>
<td>0.040</td>
<td>12</td>
<td>840</td>
</tr>
<tr>
<td>Plaster_BOARD</td>
<td>0.010</td>
<td>0.160</td>
<td>950</td>
<td>840 Inside</td>
</tr>
</tbody>
</table>

#### Construction # 3, OR12_Floor

<table>
<thead>
<tr>
<th>Description</th>
<th>Thickness</th>
<th>Lambda</th>
<th>Specific mass</th>
<th>Specific Heat</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insulation</td>
<td>0.200</td>
<td>0.040</td>
<td>10</td>
<td>1400 Outside</td>
</tr>
<tr>
<td>Insulation</td>
<td>0.200</td>
<td>0.040</td>
<td>10</td>
<td>1400</td>
</tr>
<tr>
<td>Insulation</td>
<td>0.200</td>
<td>0.040</td>
<td>10</td>
<td>1400</td>
</tr>
<tr>
<td>Insulation</td>
<td>0.200</td>
<td>0.040</td>
<td>10</td>
<td>1400</td>
</tr>
<tr>
<td>Insulation</td>
<td>0.200</td>
<td>0.040</td>
<td>10</td>
<td>1400</td>
</tr>
<tr>
<td>Concrete_slab</td>
<td>0.040</td>
<td>1.130</td>
<td>1400</td>
<td>1000</td>
</tr>
<tr>
<td>Concrete_slab</td>
<td>0.040</td>
<td>1.130</td>
<td>1400</td>
<td>1000 Inside</td>
</tr>
</tbody>
</table>

Appendix II / VII
Construction # 4, ON24_window_kozijn

<table>
<thead>
<tr>
<th>Description</th>
<th>Thickness [m]</th>
<th>Lambda [W/m.K]</th>
<th>Specific mass [kg/m²]</th>
<th>Specific Heat [J/kg.K]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dummy</td>
<td>0.038</td>
<td>0.400</td>
<td>2800</td>
<td>1470</td>
</tr>
</tbody>
</table>

Absorption | Emission | Convection [W/m²]
Top/Outside | 0.6 | 0.9 | 18.0
Bottom/Inside | 0.6 | 0.9 | 3.0

II.4 Systems

Generation

<table>
<thead>
<tr>
<th>Generator</th>
<th>Flow temperature</th>
<th>Return temperature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heating</td>
<td>Present 80.0 °C</td>
<td>60.0 °C</td>
</tr>
<tr>
<td>Cooling</td>
<td>Present 6.0 °C</td>
<td>12.0 °C</td>
</tr>
</tbody>
</table>

Distribution, heating curves

**Hot Water**

<table>
<thead>
<tr>
<th>Day</th>
<th>Night</th>
</tr>
</thead>
<tbody>
<tr>
<td>To [°C]</td>
<td>Ti [°C]</td>
</tr>
<tr>
<td>-10.0</td>
<td>45.0</td>
</tr>
<tr>
<td>12.0</td>
<td>21.0</td>
</tr>
<tr>
<td>12.1</td>
<td>18.0</td>
</tr>
<tr>
<td>30.0</td>
<td>15.0</td>
</tr>
<tr>
<td>30.0</td>
<td>15.0</td>
</tr>
<tr>
<td>30.0</td>
<td>15.0</td>
</tr>
</tbody>
</table>

**Cold Water**

<table>
<thead>
<tr>
<th>Day</th>
<th>Night</th>
</tr>
</thead>
<tbody>
<tr>
<td>To [°C]</td>
<td>Ti [°C]</td>
</tr>
<tr>
<td>-10.0</td>
<td>45.0</td>
</tr>
<tr>
<td>12.0</td>
<td>21.0</td>
</tr>
<tr>
<td>12.1</td>
<td>18.0</td>
</tr>
<tr>
<td>30.0</td>
<td>15.0</td>
</tr>
<tr>
<td>30.0</td>
<td>15.0</td>
</tr>
<tr>
<td>30.0</td>
<td>15.0</td>
</tr>
</tbody>
</table>

Air handling

No central air handling present.

Floor system

<table>
<thead>
<tr>
<th>Heating</th>
<th>Cooling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power</td>
<td>45 W</td>
</tr>
<tr>
<td>at water supply temperature [°C]</td>
<td>35.0</td>
</tr>
<tr>
<td>at water return temperature [°C]</td>
<td>27.5</td>
</tr>
<tr>
<td>at outside temperature [°C]</td>
<td>20.0</td>
</tr>
</tbody>
</table>

Temperature set points

- daytime setpoint [°C] | 20.0 | 24.0 |
- standby setpoint [°C] | 18.0 | 26.0 |

Infiltration

| Wind speed [m/s] | 0 | 3 | 6 |
| Infiltration [1/h] | 0.10 | 0.30 | 0.50 |

Windows

The window cannot be opened.

Ventilation

Ventilation flow rate in m³/h. In any listing with (off) there is no ventilation

<table>
<thead>
<tr>
<th>Daytime</th>
<th>Night time (standby)</th>
<th>Conditional night ventilation (Night cooling and heating)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supply</td>
<td>Exhaust</td>
<td>Supply</td>
</tr>
<tr>
<td>1 : 1_-_building</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
### II.5 Internal heat gains

#### People

**People**
- **271.0 W**
- **Summer clothing**
  - **0.7 CLO**
- **Winter clothing**
  - **0.9 CLO**
- **Activity**
  - **1.20 MET**

<table>
<thead>
<tr>
<th>hour</th>
<th>Mon</th>
<th>Tue</th>
<th>Wed</th>
<th>Thu</th>
<th>Fri</th>
<th>Sat</th>
<th>Sun</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>271</td>
<td>271</td>
<td>271</td>
<td>271</td>
<td>271</td>
<td>271</td>
<td>271</td>
</tr>
</tbody>
</table>

#### Devices

**Thermal power**
- **90.3 W**

<table>
<thead>
<tr>
<th>hour</th>
<th>Mon</th>
<th>Tue</th>
<th>Wed</th>
<th>Thu</th>
<th>Fri</th>
<th>Sat</th>
<th>Sun</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>90</td>
<td>90</td>
<td>90</td>
<td>90</td>
<td>90</td>
<td>90</td>
<td>90</td>
</tr>
</tbody>
</table>

**Latent part**
- **0.00 -**

**Sensible part**
- **1.00 -**

**Convective part**
- **1.00 -**

#### Lighting

**Thermal power**
- **180.6 W**

<table>
<thead>
<tr>
<th>hour</th>
<th>Mon</th>
<th>Tue</th>
<th>Wed</th>
<th>Thu</th>
<th>Fri</th>
<th>Sat</th>
<th>Sun</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.20</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>0.72</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

**Control**
- **manual**

**Latent part**
- **0.00 -**

**Sensible part**
- **1.00 -**

**Convective part**
- **0.72 -**

### II.6 Time schedules

#### Counting hours

- **0** = Not counted, **1** = Counted

Counting hours are calculated based on the start and end period of the climate file.

<table>
<thead>
<tr>
<th>hour</th>
<th>Mon</th>
<th>Tue</th>
<th>Wed</th>
<th>Thu</th>
<th>Fri</th>
<th>Sat</th>
<th>Sun</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Appendix II / VII
VARIANT 2: WIND-DEPENDENT INFILTRATION RATE ($q_{inw}$) DOUBLED

Infiltration

<table>
<thead>
<tr>
<th>Room</th>
<th>Wind speed [m/s]</th>
<th>Infiltration [1/h]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: 1_-- building</td>
<td>0</td>
<td>0.20</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>0.60</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>1.00</td>
</tr>
</tbody>
</table>

VARIANT 3: G-VALUE WINDOWS ($q_{win}$) INCREASED

Transparent constructions

<table>
<thead>
<tr>
<th>Constr Ref #</th>
<th>Description on</th>
<th>Absorption [-]</th>
<th>Emission [-]</th>
<th>Convection [W/m2]</th>
<th>U [-]</th>
<th>ZTA [W/m2/K]</th>
<th>CF [-]</th>
<th>Sun Protection Placing Circuit</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>ON4_window_glas</td>
<td>1</td>
<td>0.100</td>
<td>0.90</td>
<td>23.00</td>
<td>1.10</td>
<td>0.691</td>
<td>0.05</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>0.100</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

VARIANT 4: INTERNAL GAINS ($q_{int}$) DOUBLED

People

<table>
<thead>
<tr>
<th>People</th>
<th>542.0 W</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summer clothing</td>
<td>0.7 CLO</td>
</tr>
<tr>
<td>Winter clothing</td>
<td>0.9 CLO</td>
</tr>
</tbody>
</table>

Activity

<table>
<thead>
<tr>
<th>Mon</th>
<th>Tue</th>
<th>Wed</th>
<th>Thu</th>
<th>Fri</th>
<th>Sat</th>
<th>Sun</th>
<th>Hol</th>
<th>Nat.Hol</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Devices

Thermal power

<table>
<thead>
<tr>
<th>Mon</th>
<th>Tue</th>
<th>Wed</th>
<th>Thu</th>
<th>Fri</th>
<th>Sat</th>
<th>Sun</th>
<th>Hol</th>
<th>Nat.Hol</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

VARIANT 5: TIGHTER TEMPERATURE SET POINTS ($T_{set}$)

Temperature set points

- daytime set point [°C] | 21.0 | 23.0 |
- standby set point [°C] | 19.0 | 25.0 |

Appendix II / VII
APPENDIX III: EMULATED DATA (VABI RESULTS)

III.1 Input settings

- Internal heat gains [W]

![Internal heat gains for a winter (left) and summer week (right) of the NEN 5060 E dataset](image1)

*Figure III.1: Internal heat gains for a winter (left) and summer week (right) of the NEN 5060 E dataset*

- Zone air temperature set points [°C]

![Temperature set points for a winter (left) and summer week (right) of the NEN 5060 E dataset](image2)

*Figure III.2: Temperature set points for a winter (left) and summer week (right) of the NEN 5060 E dataset*

- Floor system supply water temperature [°C]

![Supply water temperature set points according to the ambient temperature based heating curve](image3)

*Figure III.3: Supply water temperature set points according to the ambient temperature based heating curve*
III.2 Output NEN 5060 E

- Solar gains [W]

Figure III.4: Solar gains for a winter (left) and summer week (right) of the NEN 5060 E dataset

- Floor system supply water temperature [°C]

Figure III.5: Floor system supply water temperature for a winter (left) and summer week (right) of the NEN 5060 E dataset (case variant 1)

- Floor system return water temperature [°C]

Figure III.6: Floor system return water temperature for a winter (left) and summer week (right) of the NEN 5060 E dataset (case variant 1)

- Indoor air temperature [°C]

Figure III.7: Indoor air temperature for a winter (left) and summer week (right) of the NEN 5060 E dataset

Appendix III / VII
• Floor surface temperature [°C]

Figure III.8: Floor surface temperature for a winter (left) and summer week (right) of the NEN 5060 E dataset

• Heating and cooling power [W]

Figure III.9: Heating and cooling power for a winter (left) and summer week (right) of the NEN 5060 E dataset

• Annual heating and cooling demand [MWh]

Figure III.10: Annual heating and cooling demand for the NEN 5060 E dataset
III.3 Output NEN 5060 ref TO5

- Solar gains [W]

Figure III.11: Solar gains for a winter (left) and summer week (right) of the NEN 5060 ref TO5 dataset

- Floor system supply water temperature [°C]

Figure III.12: Floor system supply water temperature for a winter (left) and summer week (right) of the NEN 5060 ref TO5 dataset (case variant 1)

- Floor system return water temperature [°C]

Figure III.13: Floor system return water temperature for a winter (left) and summer week (right) of the NEN 5060 ref TO5 dataset (case variant 1)

- Indoor air temperature [°C]

Figure III.14: Indoor air temperature for a winter (left) and summer week (right) of the NEN 5060 ref TO5 dataset
• Floor surface temperature [°C]

![Floor surface temperature](image)

*Figure III.15: Floor surface temperature for a winter (left) and summer week (right) of the NEN 5060 ref TO5 dataset*

• Heating and cooling power [W]

![Heating and cooling power](image)

*Figure III.16: Heating and cooling power for a winter (left) and summer week (right) of the NEN 5060 ref TO5 dataset*

• Annual heating and cooling demand [MWh]

![Annual heating and cooling demand](image)

*Figure III.17: Annual heating and cooling demand for the NEN 5060 ref TO5 dataset*
APPENDIX IV: LUMPED RC BUILDING NETWORKS

G3C3A1 network

The ODES of the G3C3A1 network are:

\[
\begin{align*}
\frac{d\hat{T}_{\text{env}}}{dt} &= \frac{G_{\text{env}}}{C_{\text{env}}} (\hat{T}_i - \hat{T}_{\text{env}}) \\
\frac{d\hat{T}_i}{dt} &= \frac{G_{\text{fast}}}{C_i} (T_e - \hat{T}_i) + \frac{G_{\text{env}}}{C_i} (\hat{T}_{\text{env}} - \hat{T}_i) + \frac{G_{\text{fl}}}{C_i} (\hat{T}_{\text{fl}} - \hat{T}_i) + \frac{A_{\text{win}}}{C_i} \text{Irrad} + \frac{1}{C_i} Q_{\text{int}} \\
\frac{d\hat{T}_{\text{fl}}}{dt} &= \frac{G_{\text{fl}}}{C_{\text{fl}}} (\hat{T}_i - \hat{T}_{\text{fl}}) + \frac{1}{C_{\text{fl}}} P
\end{align*}
\]

The state space matrices of the G3C3A1 network are:

\[
A = \begin{bmatrix}
\frac{G_{\text{env}}}{C_{\text{env}}} & \frac{G_{\text{env}}}{C_{\text{env}}} & 0 \\
\frac{G_{\text{fast}}}{C_i} & \frac{\hat{T}_i}{C_i} - \frac{G_{\text{env}}}{C_i} - \frac{G_{\text{fl}}}{C_i} & \frac{G_{\text{fl}}}{C_i} \\
0 & \frac{G_{\text{fl}}}{C_{\text{fl}}} & \frac{1}{C_{\text{fl}}}
\end{bmatrix}
\]

\[
B = \begin{bmatrix}
\frac{G_{\text{env}}}{C_i} & \frac{A_{\text{win}}}{C_i} & \frac{1}{C_i} & 0 \\
0 & 0 & 0 & \frac{1}{G_{\text{fl}}}
\end{bmatrix}
\]

\[
C = \begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

\[
D = 0
\]

With state matrix \( \dot{x}_k \):

\[
\dot{x}_k = \begin{bmatrix} T_{\text{env}} & T_i & T_{\text{fl}} \end{bmatrix}
\]

And input matrix \( u_k \):

\[
u_k = \begin{bmatrix} T_e & \text{Irrad} & Q_{\text{int}} & P \end{bmatrix}
\]
The ODES of the G4C4A1 network are:

\[
\frac{dT_{env}}{dt} = \frac{G_{env}}{C_{env}} (T_i - T_{env})
\]

\[
\frac{dT_i}{dt} = \frac{G_{fast}}{C_i} (T_e - T_i) + \frac{G_{env}}{C_i} (T_{env} - T_i) + \frac{G_{int}}{C_i} (T_{int} - T_i) + \frac{G_{fl}}{C_i} (T_{fl} - T_i) + \frac{1}{C_i} Q_{int}
\]

\[
\frac{dT_{int}}{dt} = \frac{G_{int}}{C_{int}} (T_i - T_{int}) + \frac{A_{win2}}{C_{int}} T_{rad}
\]

\[
\frac{dT_{fl}}{dt} = \frac{G_{fl}}{C_{fl}} (T_i - T_{fl}) + \frac{1}{C_{fl}} Q
\]

The state space matrices of the G4C4A1 network are:

\[
A = \begin{bmatrix}
\frac{G_{env}}{C_{env}} & \frac{G_{env}}{C_{env}} & 0 & 0 \\
-\frac{G_{fast}}{C_i} & -\frac{G_{env}}{C_i} & -\frac{G_{int}}{C_i} & -\frac{G_{fl}}{C_i} \\
0 & \frac{G_{int}}{C_{int}} & -\frac{G_{int}}{C_{int}} & 0 \\
0 & \frac{G_{fl}}{C_{fl}} & 0 & -\frac{G_{fl}}{C_{fl}}
\end{bmatrix}
\]

\[
B = \begin{bmatrix}
\frac{G_{env}}{C_i} & 0 & \frac{1}{C_i} & 0 \\
0 & A_{win2} & 0 & 0 \\
0 & 0 & 0 & \frac{1}{C_{fl}}
\end{bmatrix}
\]

\[
C = \begin{bmatrix}
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

\[
D = 0
\]

With state matrix \( \dot{x}_k \):

\[
\dot{x}_k = [T_{env} \ T_i \ T_{int} \ T_{fl}]
\]

And input matrix \( u_k \):

\[
u_k = \begin{bmatrix}
T_e & T_{rad} & Q_{int} & P
\end{bmatrix}
\]
The ODES of the G4C4A2 network are:

\[
\begin{align*}
\frac{d\hat{T}_{\text{env}}}{dt} &= \frac{g_{\text{env}}}{c_{\text{env}}} (\hat{T}_i - \hat{T}_{\text{env}}) \\
\frac{d\hat{T}_i}{dt} &= \frac{g_{\text{fast}}}{c_i} (\hat{T}_e - \hat{T}_i) + \frac{g_{\text{env}}}{c_i} (\hat{T}_{\text{env}} - \hat{T}_i) + \frac{g_{\text{int}}}{c_i} (\hat{T}_{\text{int}} - \hat{T}_i) + \frac{g_{f1}}{c_i} (\hat{T}_{f1} - \hat{T}_i) + \frac{A_{\text{win}}}{c_i} \overline{\text{Irrad}} + \frac{1}{c_i} Q_{\text{int}} \\
\frac{d\hat{T}_{\text{int}}}{dt} &= \frac{g_{f1}}{c_{f1}} (\hat{T}_i - \hat{T}_{\text{int}}) + \frac{A_{\text{win}2}}{c_{f1}} \overline{\text{Irrad}} \\
\frac{d\hat{T}_{f1}}{dt} &= \frac{g_{f1}}{c_{f1}} (\hat{T}_i - \hat{T}_{f1}) + \frac{1}{c_{f1}} P
\end{align*}
\]

The state space matrices of the G4C4A2 network are:

\[
A = \begin{bmatrix}
\frac{-g_{\text{env}}}{c_{\text{env}}} & \frac{g_{\text{env}}}{c_{\text{env}}} & 0 & 0 \\
\frac{g_{\text{env}}}{c_i} & \frac{-g_{\text{fast}} - g_{\text{env}} - g_{\text{int}} - g_{f1}}{c_i} & \frac{g_{\text{int}}}{c_i} & \frac{g_{f1}}{c_i} \\
0 & \frac{g_{\text{int}}}{c_{f1}} & \frac{-g_{\text{int}}}{c_{f1}} & 0 \\
0 & \frac{g_{f1}}{c_{f1}} & 0 & \frac{-g_{f1}}{c_{f1}}
\end{bmatrix}
\]

\[
B = \begin{bmatrix}
\frac{g_{\text{env}}}{c_i} & \frac{A_{\text{win}}}{c_i} & \frac{1}{c_i} & 0 \\
0 & \frac{A_{\text{win}2}}{c_{f1}} & 0 & 0 \\
0 & 0 & 0 & \frac{1}{c_{f1}}
\end{bmatrix}
\]

\[
C = \begin{bmatrix}
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

\[
D = 0
\]

With state matrix \( \dot{x}_k \):

\[
\dot{x}_k = [\hat{T}_{\text{env}} \ T_i \ T_{\text{int}} \ T_{f1}]
\]

And input matrix \( u_k \):

\[
u_k = [T_e \ \overline{\text{Irrad}} \ Q_{\text{int}} \ P]
\]
The ODES of the G7C6A2 network are:

\[
\begin{align*}
\frac{dT_{env2}}{dt} &= \frac{G_{env,2}}{C_{env2}} (T_{env} - T_{env2}) + \frac{G_{env,3}}{C_{env2}} (T_e - T_{env2}) \\
\frac{dT_{env}}{dt} &= \frac{G_{env,2}}{C_{env}} (T_{env2} - T_{env}) + \frac{G_{env}}{C_{env}} (T_i - T_{env}) \\
\frac{dT_i}{dt} &= \frac{G_{fast}}{C_i} (T_e - T_i) + \frac{G_{int}}{C_i} (T_{int} - T_i) + \frac{G_{f1}}{C_i} (T_{f1} - T_i) + \frac{A_{win}}{C_i} Irrad + \frac{1}{C_i} Q_{int} \\
\frac{dT_{int}}{dt} &= \frac{G_{int}}{C_{int}} (T_i - T_{int}) + \frac{A_{win}^2}{C_{int}} Irrad \\
\frac{dT_{f1}}{dt} &= \frac{G_{f1}}{C_{f1}} (T_i - T_{f1}) + \frac{G_{f12}}{C_{f1}} (T_{f12} - T_{f1}) + \frac{1}{C_{f1}} P \\
\frac{dT_{f12}}{dt} &= (T_{f1} - T_{f12})
\end{align*}
\]

The state space matrices of the G7C6A2 network are:

\[
A = \begin{bmatrix}
G_{env,3} & -G_{env,2} & 0 & 0 & 0 & 0 \\
-G_{env,3} & \frac{C_{env} G_{env,2}}{C_{env2}} & \frac{G_{env}}{C_{env}} & 0 & 0 & 0 \\
\frac{G_{env}}{C_{i}} & \frac{G_{int}}{C_{i}} & \frac{G_{f1}}{C_{i}} & 0 & 0 & 0 \\
0 & \frac{G_{int}}{C_{int}} & \frac{G_{f12}}{C_{int}} & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{G_{f12}}{C_{f12}} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{G_{f12}}{C_{f12}} & 0
\end{bmatrix}
\]

\[
B = \begin{bmatrix}
\frac{G_{env,3}}{C_{env2}} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
\frac{G_{fast}}{C_i} & \frac{A_{win}}{C_i} & \frac{1}{C_i} & 0 \\
0 & \frac{A_{win}^2}{C_{int}} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{C_{f1}} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{1}{C_{f1}} & 0
\end{bmatrix}
\]

Appendix IV / VII
\[ C = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \]

\[ D = 0 \]

With state matrix \( \dot{x}_k \):
\[
\dot{x}_k = [T_{env2} \ T_{env} \ T_l \ T_{int} \ T_{f_l} \ T_{f_{l2}}]
\]

And input matrix \( u_k \):
\[
u_k = [\tau_c \ \bar{irrad} \ Q_{int} \ P]
\]
APPENDIX V: SYSTEM IDENTIFICATION SETTINGS

V.1 Genetic Algorithm

PopulationType: [ 'bitstring' | 'custom' | {'doubleVector'} ]
PopInitRange: [ matrix | {{[-10;10]}} ]
PopulationSize: [ positive scalar ]
EliteCount: [ positive scalar | {0.05*PopulationSize} ]
CrossoverFraction: [ positive scalar | {0.8} ]
ParetoFraction: [ positive scalar | {0.35} ]
MigrationDirection: [ 'both' | {'forward'} ]
MigrationInterval: [ positive scalar | {20} ]
MigrationFraction: [ positive scalar | {0.2} ]

Generations: [ positive scalar ]
TimeLimit: [ positive scalar | {Inf} ]
FitnessLimit: [ scalar | {Inf} ]
StallGenLimit: [ positive scalar ]
StallTest: [ 'geometricWeighted' | {'averageChange'} ]
StallTimeLimit: [ positive scalar | {Inf} ]
TolFun: [ positive scalar ]
TolCon: [ positive scalar | {1e-6} ]

InitialPopulation: [ matrix | {} ]
InitialScores: [ column vector | {} ]

NonlinConAlgorithm: [ 'penalty' | {'auglag'} ]
InitialPenalty: [ positive scalar | {10} ]
PenaltyFactor: [ positive scalar | {100} ]

CreationFcn: [ function_handle | @gacreationuniform ]
FitnessScalingFcn: [ function_handle | @fitscalingshiftlinear ]
SelectionFcn: [ function_handle | @selectionremainder ]
CrossoverFcn: [ function_handle | @crossoversinglepoint ]
MutationFcn: [ function_handle | @mutationuniform ]
DistanceMeasureFcn: [ function_handle | @distancecrowding ]
HybridFcn: [ @fminsearch | @patternsearch | @fminunc | @fmincon | {} ]
Appendix V / VII

PlotFcns: \[ \text{function_handle} | \text{@gaplotbestf} | \text{@gaplotbestindiv} | \text{@gaplotdistance} | \text{@gaplotexpectation} | \text{@gaplotgenealogy} | \text{@gaplotselection} | \text{@gaplotrange} | \text{@gaplotscorediversity} | \text{@gaplotscores} | \text{@gaplotstopping} \]

\[ \text{@gaplotmaxconstr} | \text{@gaplotrankhist} | \text{@gaplotpareto} | \text{@gaplotspread} | \text{@gaplotparetoddistance} \]

\[
\text{PlotInterval: [} \{\text{"positive scalar"}\} | \{1\} \]
\text{Vectorized: [} \text{'on'} | \{\text{"off"}\} \]
\text{UseParallel: [} \text{logical scalar} | \text{true} | \{\text{false}\} \]

V.2 PatternSearch

\text{UseParallel: [} \text{logical scalar} | \text{true} | \{\text{false}\} \]

\[
\text{ToMesh: [} \text{positive scalar} | \{1e-6\} \]
\text{ToCon: [} \text{positive scalar} | \{1e-6\} \]
\text{ToX: [} \text{positive scalar} | \{1e-6\} \]
\text{ToFun: [} \text{positive scalar} | \{1e-6\} \]
\text{ToBind: [} \text{positive scalar} | \{1e-3\} \]
\text{MaxIter: [} \text{positive scalar} | \{100*numberOfVariables\} \]
\text{MaxFunEvals: [} \text{positive scalar} | \{2000*numberOfVariables\} \]
\text{TimeLimit: [} \text{positive scalar} | \{Inf\} \]
\text{MeshContraction: [} \text{positive scalar} | \{0.5\} \]
\text{MeshExpansion: [} \text{positive scalar} | \{2.0\} \]
\text{MeshAccelerator: [} \text{on} | \{\text{off}\} \]
\text{MeshRotate: [} \text{off} | \{\text{on}\} \]
\text{InitialMeshSize: [} \text{positive scalar} | \{1.0\} \]
\text{ScaleMesh: [} \text{off} | \{\text{on}\} \]
\text{MaxMeshSize: [} \text{positive scalar} | \{Inf\} \]
\text{InitialPenalty: [} \text{positive scalar} | \{10\} \]
\text{PenaltyFactor: [} \text{positive scalar} | \{100\} \]

\[
\text{PollMethod: [} \text{MADSPositiveBasisNp1} | \text{MADSPositiveBasis2N} | \text{GPSPositiveBasisNp1} | \{\text{GPSPositiveBasis2N}\} \]
\text{CompletePoll: [} \text{on} | \{\text{off}\} \]
\text{PollingOrder: [} \text{Random} | \text{Success} | \{\text{Consecutive}\} \]
\text{SearchMethod: [} \text{function_handle} | \text{@MADSPositiveBasisNp1} | \text{@MADSPositiveBasis2N} | \text{@GPSPositiveBasisNp1} | \text{@GPSPositiveBasis2N} | \text{@GSSPositiveBasisNp1} | \text{@GSSPositiveBasis2N} | \text{@searchga} | \text{@searchlhs} | \text{@searchneldermead} | \{[]\} \]
\text{CompleteSearch: [} \text{on} | \{\text{off}\} \]

Appendix V / VII
Appendix V/ VII

Display: [ off | iter | diagnose | {final} ]

OutputFcns: [ function_handle | {[]} ]

PlotFcns: [ function_handle | @psplotbestf |
            @psplotmeshsize | @psplotfuncount |
            @psplotbestx | {[]} ]

PlotInterval: [ positive scalar | {1} ]

Cache: [ on | {off} ]
CacheSize: [ positive scalar | {1e4} ]
CacheTol: [ positive scalar | {eps} ]

Vectorized: [ on | {off} ]

UseParallel: [ logical scalar | true | {false} ]

Display: [ off | iter | iter-detailed | notify | notify-detailed | final | final-detailed ]

V.3 Fmincon

MaxFunEvals: [ positive scalar ]
MaxIter: [ positive scalar ]
TolFun: [ positive scalar ]
TolX: [ positive scalar ]
FunValCheck: [ on | {off} ]
OutputFcn: [ function | {[]} ]
PlotFcns: [ function | {[]} ]
Algorithm: [ active-set | interior-point | interior-point-convex | levenberg-marquardt | ...
           simplex | sqp | trust-region-dogleg | trust-region-reflective ]

AlwaysHonorConstraints: [ none | {bounds} ]
DerivativeCheck: [ on | {off} ]
Diagnostics: [ on | {off} ]
DiffMaxChange: [ positive scalar | {Inf} ]
DiffMinChange: [ positive scalar | {0} ]
FinDiffRelStep: [ positive vector | positive scalar | {[]} ]
FinDiffType: [ {forward} | central ]
GoalsExactAchieve: [ positive scalar | {0} ]
GradConstr: [ on | {off} ]
GradObj: [ on | {off} ]
HessFcn: [ function | {[]} ]
Hessian: [ user-supplied | bfgs | lbfgs | fin-diff-grads | on | off ]
HessMult: [ function | {[]} ]
HessPattern: [ sparse matrix | {sparse(ones(numberOfVariables))} ]
HessUpdate: [ dfp | steepest | {bfgs} ]
InitBarrierParam: [ positive scalar | {0.1} ]
InitialHessType: [ identity | {scaled-identity} | user-supplied ]
InitialHessMatrix: [ scalar | vector | {[]} ]
InitTrustRegionRadius: [ positive scalar | {sqrt(numberOfVariables)} ]
Jacobian: [ on | {off} ]
JacobMult: [ function | {[]} ]
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>JacobPattern:</td>
<td>[ sparse matrix</td>
</tr>
<tr>
<td>LargeScale:</td>
<td>[ on</td>
</tr>
<tr>
<td>MaxNodes:</td>
<td>[ positive scalar</td>
</tr>
<tr>
<td>MaxPCGIter:</td>
<td>[ positive scalar</td>
</tr>
<tr>
<td>MaxProjCGIter:</td>
<td>[ positive scalar</td>
</tr>
<tr>
<td>MaxSQPIter:</td>
<td>[ positive scalar</td>
</tr>
<tr>
<td>MaxTime:</td>
<td>[ positive scalar</td>
</tr>
<tr>
<td>MeritFunction:</td>
<td>[ singleobj</td>
</tr>
<tr>
<td>MinAbsMax:</td>
<td>[ positive scalar</td>
</tr>
<tr>
<td>ObjectiveLimit:</td>
<td>[ scalar</td>
</tr>
<tr>
<td>PrecondBandWidth:</td>
<td>[ positive scalar</td>
</tr>
<tr>
<td>RelLineSrchBnd:</td>
<td>[ positive scalar</td>
</tr>
<tr>
<td>RelLineSrchBndDuration:</td>
<td>[ positive scalar</td>
</tr>
<tr>
<td>ScaleProblem:</td>
<td>[ none</td>
</tr>
<tr>
<td>Simplex:</td>
<td>[ on</td>
</tr>
<tr>
<td>SubproblemAlgorithm:</td>
<td>[ cg</td>
</tr>
<tr>
<td>TolCon:</td>
<td>[ positive scalar ]</td>
</tr>
<tr>
<td>TolConSQP:</td>
<td>[ positive scalar</td>
</tr>
<tr>
<td>TolPCG:</td>
<td>[ positive scalar</td>
</tr>
<tr>
<td>TolProjCG:</td>
<td>[ positive scalar</td>
</tr>
<tr>
<td>TolProjCGAbs:</td>
<td>[ positive scalar</td>
</tr>
<tr>
<td>TypicalX:</td>
<td>[ vector</td>
</tr>
<tr>
<td>UseParallel:</td>
<td>[ logical scalar</td>
</tr>
</tbody>
</table>
APPENDIX VI: RESULTS DIFFERENT LUMPED RC MODEL CONFIGURATIONS

VI.1: Results: Perez direct/diffuse split model NEN 5060 E data set

- Solar irradiance on west vertical wall

\[
I_{rad} = I_{win} \times A_{win} \times g_{win}
\]

Case variants 1, 2, 4 and 5:

\[
Q_{solar, Perez} = \frac{I_{rad}}{A_{win} \times g_{win}} \times 10.8 \times 0.400 = 4.32 \times I_{rad}
\]

Case variants 3:

\[
Q_{solar, Perez} = \frac{I_{rad}}{A_{win} \times g_{win}} \times 10.8 \times 0.691 = 7.46 \times I_{rad}
\]

VI.2: Results: Perez direct/diffuse split model NEN 5060 ref TO5 data set

- Solar irradiance on west vertical wall

\[
I_{rad} = I_{win} \times A_{win} \times g_{win}
\]

Figure VI.1: Solar irradiance on window area calculated by Perez model for a winter (left) and summer week (right) of the NEN 5060 E dataset

Figure VI.2: Solar gains calculated by Perez model for a winter (left) and summer week (right) of the NEN 5060 E dataset

Figure VI.3: Solar irradiance on window area calculated by Perez model for a winter (left) and summer week (right) of the NEN 5060 ref TO5 dataset
Solar gains calculated by Perez solar inclination model

\[ Q_{\text{Solar,Perez}} = \overline{I_{\text{rad}}} \times A_{\text{win}} \times g_{\text{win}} \]

Case variants 1,2, 4 and 5:

\[ Q_{\text{Solar,Perez}} = \overline{I_{\text{rad}}} \times 10.8 \times 0.400 = 4.32 \times \overline{I_{\text{rad}}} \]

Case variants 3:

\[ Q_{\text{Solar,Perez}} = \overline{I_{\text{rad}}} \times 10.8 \times 0.691 = 7.46 \times \overline{I_{\text{rad}}} \]

Figure VI.4: Solar gains calculated by Perez model for a winter (left) and summer week (right) of the NEN 5060 ref TO5 dataset
VI.3: Performance metrics lumped RC building model

- **G3C3A1 network**

<table>
<thead>
<tr>
<th>Table VI.1: Performance metrics identification data</th>
<th>Table VI.2: Performance metrics validation data</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Variant</strong></td>
<td><strong>MBE_h [$^{\circ}$C]</strong></td>
</tr>
<tr>
<td>1</td>
<td>0.01</td>
</tr>
<tr>
<td>2</td>
<td>0.02</td>
</tr>
<tr>
<td>3</td>
<td>0.03</td>
</tr>
<tr>
<td>4</td>
<td>0.01</td>
</tr>
<tr>
<td>5</td>
<td>0.02</td>
</tr>
</tbody>
</table>

- **G4C4A1 network**

<table>
<thead>
<tr>
<th>Table VI.3: Performance metrics identification data</th>
<th>Table VI.4: Performance metrics validation data</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Variant</strong></td>
<td><strong>MBE_h [$^{\circ}$C]</strong></td>
</tr>
<tr>
<td>1</td>
<td>0.02</td>
</tr>
<tr>
<td>2</td>
<td>0.02</td>
</tr>
<tr>
<td>3</td>
<td>0.03</td>
</tr>
<tr>
<td>4</td>
<td>0.02</td>
</tr>
<tr>
<td>5</td>
<td>0.02</td>
</tr>
</tbody>
</table>

- **G4C4A2 network**

<table>
<thead>
<tr>
<th>Table VI.5: Performance metrics identification data</th>
<th>Table VI.6: Performance metrics validation data</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Variant</strong></td>
<td><strong>MBE_h [$^{\circ}$C]</strong></td>
</tr>
<tr>
<td>1</td>
<td>0.02</td>
</tr>
<tr>
<td>2</td>
<td>0.02</td>
</tr>
<tr>
<td>3</td>
<td>0.03</td>
</tr>
<tr>
<td>4</td>
<td>0.02</td>
</tr>
<tr>
<td>5</td>
<td>0.02</td>
</tr>
</tbody>
</table>

- **G7C6A1 network**

<table>
<thead>
<tr>
<th>Table VI.7: Performance metrics identification data</th>
<th>Table VI.8: Performance metrics validation data</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Variant</strong></td>
<td><strong>MBE_h [$^{\circ}$C]</strong></td>
</tr>
<tr>
<td>1</td>
<td>0.02</td>
</tr>
<tr>
<td>2</td>
<td>0.02</td>
</tr>
<tr>
<td>3</td>
<td>0.03</td>
</tr>
<tr>
<td>4</td>
<td>0.02</td>
</tr>
<tr>
<td>5</td>
<td>0.02</td>
</tr>
</tbody>
</table>

Appendix VI/ VII
VI.4: Performance metrics coupled simulation model

- **G3C3A1 model**

Table VI.9: Performance metrics identification data

<table>
<thead>
<tr>
<th>Variant</th>
<th>RMSE&lt;sub&gt;b&lt;/sub&gt; [°C]</th>
<th>NRMSE&lt;sub&gt;b&lt;/sub&gt; [%]</th>
<th>PE&lt;sub&gt;heat,a&lt;/sub&gt; [%]</th>
<th>PE&lt;sub&gt;cool,a&lt;/sub&gt; [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variant 1</td>
<td>0.36</td>
<td>5.52</td>
<td>0.33</td>
<td>6.97</td>
</tr>
<tr>
<td>Variant 2</td>
<td>0.42</td>
<td>7.46</td>
<td>2.74</td>
<td>8.64</td>
</tr>
<tr>
<td>Variant 3</td>
<td>0.52</td>
<td>5.93</td>
<td>0.32</td>
<td>4.09</td>
</tr>
<tr>
<td>Variant 4</td>
<td>0.43</td>
<td>5.72</td>
<td>4.33</td>
<td>5.38</td>
</tr>
<tr>
<td>Variant 5</td>
<td>0.32</td>
<td>5.60</td>
<td>0.47</td>
<td>5.52</td>
</tr>
</tbody>
</table>

Table VI.10: Performance metrics validation data

<table>
<thead>
<tr>
<th>Variant</th>
<th>RMSE&lt;sub&gt;b&lt;/sub&gt; [°C]</th>
<th>NRMSE&lt;sub&gt;b&lt;/sub&gt; [%]</th>
<th>PE&lt;sub&gt;heat,a&lt;/sub&gt; [%]</th>
<th>PE&lt;sub&gt;cool,a&lt;/sub&gt; [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variant 1</td>
<td>0.35</td>
<td>5.62</td>
<td>0.62</td>
<td>3.14</td>
</tr>
<tr>
<td>Variant 2</td>
<td>0.43</td>
<td>7.79</td>
<td>2.50</td>
<td>4.24</td>
</tr>
<tr>
<td>Variant 3</td>
<td>0.48</td>
<td>6.29</td>
<td>0.52</td>
<td>1.72</td>
</tr>
<tr>
<td>Variant 4</td>
<td>0.39</td>
<td>6.15</td>
<td>3.78</td>
<td>2.65</td>
</tr>
<tr>
<td>Variant 5</td>
<td>0.32</td>
<td>5.91</td>
<td>0.56</td>
<td>2.69</td>
</tr>
</tbody>
</table>

- **G4C4A1 model**

Table VI.11: Performance metrics identification data

<table>
<thead>
<tr>
<th>Variant</th>
<th>RMSE&lt;sub&gt;b&lt;/sub&gt; [°C]</th>
<th>NRMSE&lt;sub&gt;b&lt;/sub&gt; [%]</th>
<th>PE&lt;sub&gt;heat,a&lt;/sub&gt; [%]</th>
<th>PE&lt;sub&gt;cool,a&lt;/sub&gt; [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variant 1</td>
<td>0.35</td>
<td>4.49</td>
<td>1.06</td>
<td>6.93</td>
</tr>
<tr>
<td>Variant 2</td>
<td>0.41</td>
<td>7.28</td>
<td>2.64</td>
<td>8.47</td>
</tr>
<tr>
<td>Variant 3</td>
<td>0.51</td>
<td>5.78</td>
<td>0.54</td>
<td>3.64</td>
</tr>
<tr>
<td>Variant 4</td>
<td>0.42</td>
<td>4.26</td>
<td>2.53</td>
<td>4.71</td>
</tr>
<tr>
<td>Variant 5</td>
<td>0.29</td>
<td>4.63</td>
<td>1.20</td>
<td>5.63</td>
</tr>
</tbody>
</table>

Table VI.12: Performance metrics validation data

<table>
<thead>
<tr>
<th>Variant</th>
<th>RMSE&lt;sub&gt;b&lt;/sub&gt; [°C]</th>
<th>NRMSE&lt;sub&gt;b&lt;/sub&gt; [%]</th>
<th>PE&lt;sub&gt;heat,a&lt;/sub&gt; [%]</th>
<th>PE&lt;sub&gt;cool,a&lt;/sub&gt; [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variant 1</td>
<td>0.33</td>
<td>4.61</td>
<td>1.25</td>
<td>2.94</td>
</tr>
<tr>
<td>Variant 2</td>
<td>0.42</td>
<td>7.70</td>
<td>2.38</td>
<td>3.83</td>
</tr>
<tr>
<td>Variant 3</td>
<td>0.43</td>
<td>6.36</td>
<td>4.13</td>
<td>0.65</td>
</tr>
<tr>
<td>Variant 4</td>
<td>0.37</td>
<td>4.59</td>
<td>1.82</td>
<td>2.95</td>
</tr>
<tr>
<td>Variant 5</td>
<td>0.29</td>
<td>4.85</td>
<td>1.34</td>
<td>2.65</td>
</tr>
</tbody>
</table>

- **G4C4A2 model**

Table VI.13: Performance metrics identification data

<table>
<thead>
<tr>
<th>Variant</th>
<th>RMSE&lt;sub&gt;b&lt;/sub&gt; [°C]</th>
<th>NRMSE&lt;sub&gt;b&lt;/sub&gt; [%]</th>
<th>PE&lt;sub&gt;heat,a&lt;/sub&gt; [%]</th>
<th>PE&lt;sub&gt;cool,a&lt;/sub&gt; [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variant 1</td>
<td>0.37</td>
<td>4.65</td>
<td>1.39</td>
<td>8.42</td>
</tr>
<tr>
<td>Variant 2</td>
<td>0.41</td>
<td>6.89</td>
<td>3.73</td>
<td>8.75</td>
</tr>
<tr>
<td>Variant 3</td>
<td>0.53</td>
<td>5.34</td>
<td>0.82</td>
<td>6.89</td>
</tr>
<tr>
<td>Variant 4</td>
<td>0.41</td>
<td>4.22</td>
<td>2.07</td>
<td>6.36</td>
</tr>
<tr>
<td>Variant 5</td>
<td>0.32</td>
<td>4.91</td>
<td>1.39</td>
<td>7.27</td>
</tr>
</tbody>
</table>

Table VI.14: Performance metrics validation data

<table>
<thead>
<tr>
<th>Variant</th>
<th>RMSE&lt;sub&gt;b&lt;/sub&gt; [°C]</th>
<th>NRMSE&lt;sub&gt;b&lt;/sub&gt; [%]</th>
<th>PE&lt;sub&gt;heat,a&lt;/sub&gt; [%]</th>
<th>PE&lt;sub&gt;cool,a&lt;/sub&gt; [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variant 1</td>
<td>0.36</td>
<td>4.81</td>
<td>1.51</td>
<td>4.60</td>
</tr>
<tr>
<td>Variant 2</td>
<td>0.43</td>
<td>6.96</td>
<td>3.47</td>
<td>6.77</td>
</tr>
<tr>
<td>Variant 3</td>
<td>0.51</td>
<td>5.84</td>
<td>1.32</td>
<td>4.45</td>
</tr>
<tr>
<td>Variant 4</td>
<td>0.37</td>
<td>4.51</td>
<td>1.40</td>
<td>4.59</td>
</tr>
<tr>
<td>Variant 5</td>
<td>0.33</td>
<td>4.99</td>
<td>1.39</td>
<td>4.40</td>
</tr>
</tbody>
</table>

- **G7C6A1 model**

Not investigated, because the performance metrics of the G7C6A1 network are equal to the metrics of the G4C4A2 model
VI.5 Cross-correlation coefficients

G3C3A1 model

Figure VI.5 Cross-correlations between G3C3A1 network inputs and residual for validation data set
Figure VI.6 Cross-correlations between G4C4A1 network inputs and residual for validation data set
Appendix VI/ VII

G4C4A2 model

- Variant 1
- Variant 2
- Variant 3
- Variant 4

Figure VI.7 Cross-correlations between G4C4A2 network inputs and residual for validation data set
Figure VI.8 Cross-correlations between G7C6A2 network inputs and residual for validation data set
Appendix VII: Matlab and Simulink model scripts

Outline appendix VII

- VII.1 Calculate solar irradiance on window area - Perez solar inclination model ($\Delta t = 3600$ s)

1. Calculate solar position as function of location and time
2. Calculate direct and diffuse irradiation on inclined surface from solar position, direct normal & diffuse & solar on the horizontal and window elevation & azimuth.
3. Sum diffuse and direct irradiation on inclined surface to calculate total solar irradiance on inclined surface

- VII.2 Inverse modeling process - lumped RC building model ($\Delta t = 3600$ s)

1. Set input and output (from virtual case study building ) and parameter search space
2. Start three-stage inverse modeling approach
3. Calculate objective function value for each combination of parameters defined by algorithm
4. Estimate $T_I$ and $T_B$ for identified combination of parameters

- VII.3 Determine $G_{pipes}$ - lumped RC building model ($\Delta t = 3600$ s)

<table>
<thead>
<tr>
<th>Virtual case study building:</th>
<th>Known:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_{wet}$</td>
<td>$m_w$</td>
</tr>
<tr>
<td></td>
<td>$T_{wsp}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Lumped RC building model:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_k(t) = \begin{bmatrix} T_e \ Irrad \ Q_{net} \ P \end{bmatrix}$</td>
</tr>
</tbody>
</table>

*‘Location’ is latitude and longitude of geographic location, ‘Time’ is hour and day of the year

Figure VII.1: schematic diagram of Perez model

Figure VII.2: Inputs, system and output of lumped RC building model for identification of its parameters

Figure VII.3: Inputs and system of lumped RC building model for estimation of $T_I$ and $T_B$

Figure VII.4: Data used for identification of $G_{pipes}$ (black)
VII.4 Estimate indoor air temperature and heating and cooling demands – coupled simulation model ($\Delta t = 60$ s)

Figure VII.5: Schematic diagram of coupled simulation model in Simulink with its inputs, variables and outputs

1. Simulink model – entire coupled simulation model
2. Simulink model – control block part
3. Set input and lumped RC building model; start estimation with Simulink model

VII.5 Data processing

1. Determine output with $\Delta t = 3600$
2. Cross-correlation test
VII.1 Calculate solar irradiance on window area - Perez solar inclination model \((\Delta t = 3600 \text{ s})\)

**Perez direct/diffuse split model - calculate solar position as function of location and time \((1/3)\)**

```matlab
function [elevat,azimuth,cosinc]=solposf(ihour,iday,surfpos,geopos); % Calculation of the solar position (elevation and azimuth)
% [elevat,azimuth,cosinc]=solposf(ihour,iday,surfpos,geopos);
% 
% geopos = [Local latitude (in degrees),Local longitude (degrees),Local Standard
% time Meridian (in degrees east of Greenwich)], e.g. De Bilt:geopos=[52.1,5+11/60,15]
% surfpos = [angle between surface and horizontal (degrees),azimuth of surface
% with respect to south (degrees)], e.g. vertical:surfpos(1)=90;east:surfpos(2)=-90
% iday = number of day starting with 1 the 1st of january (1<iday<365)
% ihour = the local standard time (in hours) (1<ihour<24)
% elevat = solar elevation (degrees). Before sunrise and after sunset elevation=0
% azimuth = solar azimuth (degrees). Before sunrise and after sunset azimuth=0
% cosinc = cosinus of angle between solar rays and surface normal (incident angle)
% 
% Author: Martin de Wit 22-May-1998

rad=pi/180;
LAT=geopos(1);
LON=geopos(2);
LSM=geopos(3);
beta=surfpos(1)*rad;
gamma=surfpos(2)*rad;
LST=ihour; %Local standard time
L=LAT*rad;
l=0;
s=size(ihour);
if s(1)==1
  m=s(2);
elevat=zeros(1,m*length(iday));
azimuth=zeros(1,m*length(iday));
cosinc=zeros(1,m*length(iday));
else
  m=s(1);
elevat=zeros(m*length(iday),1);
azimuth=zeros(m*length(iday),1);
cosinc=zeros(m*length(iday),1);
end
for day=iday
  l=l+1;
  theta=2*pi*(day-1)/365.25;
  el=4.901+0.033*sin(-0.031+theta)+theta; % longitude
  delta=asin(sin(23.442*rad)*sin(el)); % declination
  q1=tan(4.901+theta);
  q2=cos(23.442*rad)*tan(el); %tan(right ascension)
  ET=(atan((q1-q2)./(q1*q2+1)))*4/rad; % equation of time
  %a1=sin(L)*sin(delta);
  %a2=cos(L)*cos(delta);
  %hh=acos(-a1./a2);
  %sunr=(12-hh/(15*rad)-ET/60+(4/60)*(LSM-LON)); % sunrise
  %suns=(12+hh/(15*rad)-ET/60+(4/60)*(LSM-LON)); % sunset
  AST=LST+ET/60-(4/60)*(LSM-LON);
  h=(AST-12)*15*rad;
  alpha=acos( cos(L)*cos(delta)*cos(h) + sin(L)*sin(delta) ) ;
  phi=acos( (cos(alpha)*sin(L)-sin(delta))./(cos(alpha)*cos(L)) ).*sign(h) ;
  gam=phi-gamma;
  caicos(alpha).*cos(gam)*sin(beta)+sin(alpha)*cos(beta);
  cai=caic>0).*cai;
  k=find(alpha<0);
  alpha(k)=0*alpha(k);
  phi(k)=0*phi(k);
  elevat(1+(l-1)*m:l*m)=alpha/rad;
  azimuth(1+(l-1)*m:l*m)=phi/rad;
  cosinc(1+(l-1)*m:l*m)=cai;
end; %day
```

Appendix VII/ VII
Perez direct/diffuse split model - calculate direct and diffuse irradiation on inclined surface from solar position, direct normal & diffuse & solar on the horizontal and window elevation & azimuth (2/3)

function [soldir,soldif]=irradf(soldirn,soldif0,elevat,cosinc,iday,surfpos,albedo);
% Calculation of the irradiance on a inclined surface from the beam radiation and the diffuse radiation on a horizontal surface using the Perez anisotropic sky.
%
% [soldir,soldif]=irradf(soldirn,soldif0,elevat,cosinc,iday,surfpos,albedo);
% soldirn = direct (=beam)solar irradiance (W/m2)
% soldif0 = diffuse (=sky) solar irradiance on a horizontal plane(W/m2)
% elevat  = solar elevation (degrees). Before sunrise and after sunset elevation=0
% azimuth = solar azimuth (degrees). Before sunrise and after sunset azimuth=0
% cosinc  = cosinus of angle between solar rays and surface normal (incident angle)
% iday    = day of the year    (1-365)
% surfpos = [angle between surface and horizontal (degrees),azimuth of surface with respect to south (degrees)], e.g. vertical: surfpos(1)=90; east: surfpos(2)=-90
% albedo  = ground reflectivity (albedo)
% soldir  = direct solar irradiation on an inclined surface
% soldif  = diffuse solar irradiation on an inclined surface
%
% Author: Martin de Wit 22-May-1998

rad=pi/180;
beta=surfpos(1)*rad;
m=length(elevat)/length(iday);
soldir=0*elevat;
soldif=0*elevat;
% sunrise and sunset
% hh=acos(-sinlat/coslat*tan(delta))/(r*15);
% sunr=ceil(12-hh-ET/60+(4/60)*(LSM-LON)+0.5);
% suns=floor(12+hh-ET/60+(4/60)*(LSM-LON)+0.5);
% Approximation of A and C, the solid angles occupied by the circumsolar region, weighed by its average incidence on the slope and horizontal respectively.
% In the expression of diffuse on inclined surface the quotient of A/C is reduced to XIC/XIH. A=2*(1-cos(beta))*xic, C=2*(1-cos(beta))*xih
epsint=[1.056 1.253 1.586 2.134 3.23 5.98 10.08 999999];
facc1=[-0.011 -0.038 0.166 0.419 0.710 0.857 0.734 0.421;...
0.748 1.115 0.909 0.646 0.025 -0.370 -0.073 -0.661;...
-0.080 -0.109 -0.179 -0.262 -0.290 -0.279 -0.228 0.097];
facc2=[-0.048 -0.023 0.062 0.140 0.243 0.267 0.231 0.119;...
0.073 0.106 -0.021 -0.167 -0.511 -0.792 -1.180 -2.125;...
-0.024 -0.037 -0.050 -0.042 -0.004 0.076 0.199 0.446];
l=0;
for day=iday
theta=360*rad*(day-1)/365.25;
l=l+1;
% calculation of extraterrestrial radiation
Eon=1370*(1+0.033*cos(theta-360*rad*2/365.25));
for hour=1:m
Enorm=soldirn(hour+(l-1)*m);
Edif=soldif0(hour+(l-1)*m);
salt=elevat(hour+(l-1)*m)*rad;
hai=sin(salt);
cai=cosinc(hour+(l-1)*m);
if Edif>0;
% determination of zet = solar zenith angle (pi/2 - solar altitude).
zet=pi/2-salt;
% determination of inteps with eps
inteps=1;
eps=1+Enorm./Edif;
i=find(epsint>=eps);
inteps=min(i);
%eps=1+Enorm./Edif
% calculation of inverse relative air mass
airmiv=hai;
if salt<10*rad,
airmiv=hai+0.15*(salt/rad+3.885)^(-1.253);
end
% delt is "the new sky brightness parameter"
delt=Edif/[airmiv*Eon];
facc1(:,inteps);
%f2acc(2); % determination of the "new circumsolar brightness coefficient % (facc(1)) and horizon brightness coefficient (f2acc(2))"
facc=[1 delt zet]*[facc1(:,inteps),facc2(:,inteps)];
% salts=solar altitude for an inclined surface
salt=pi/2-acos(cai);
% alpha= the half-angle circumsolar region
alpha=25*rad;
xic0=0.5*(1+salts/alpha).*sin((salts+alpha)/2);
xic=(salts>alpha).*xic0+(salts>alpha).*(cai-xic0);
if salt>alpha,  
  xih=hai;
else
  xih=sin((alpha+salt)/2);
end
% determination of the diffuse radiation on an inclined surface
% Isotropic sky daarna Perez
Ed=0.5*(1+cos(beta))*Edif;
Ed=Ed+Edif*(facc*(-0.5*(1+cos(beta))+xic/xih);sin(beta)));
Ed=(Ed>0).*Ed;
% horizontal surfaces treated separately
% beta=0 : surface facing up, beta=180(pi) : surface facing down
Ed=Ed+(beta>0.001 & beta<0.001).*Edif-Ed);...
(betapi-0.001) & beta<(pi+0.001)).*Ed;
else Ed=0;
  Edif=0;
end; %end Edif>0
Ehor=cai.*Enorm+Edif;
soldif(hour+(l-1)*m)=Ed+0.5*albedo*(1-cos(beta))*Ehor;
soldir(hour+(l-1)*m)=cai*Enorm;
% total irradiation on an inclined surface
end; %hour
end % end day

Perez direct/diffuse split model - sum diffuse and direct irradiation on inclined surface to calculate total solar irradiance on inclined surface (3/3)
% Prepare for function solposf
ihour=[1:24]';
iday=[1:365]';
geopos=[52.1,5+11/60,15]; % De Bilt
surfpos=[90,180 ; 90,-90 ; 90,0 ; 90,90]; % [north; east; south; west walls]
surfpos=[90,-135 ; 90,-45 ; 90,45 ; 90,135]; % [NE; SE; SW; NW walls]
% % North wall:
% [elevat,azimuth,cosincN]=solposf(ihour,iday,surfpos(1,:),geopos);
% % East wall:
% [elevat,azimuth,cosincE]=solposf(ihour,iday,surfpos(2,:),geopos);
% % South wall:
% [elevat,azimuth,cosincS]=solposf(ihour,iday,surfpos(3,:),geopos);
% West wall:
[elevat,azimuth,cosincW]=solposf(ihour,iday,surfpos(4,:),geopos);
albedo=0.2;
% NEN 5060 E or NEN 5060 ref TO5
for y=5060
  eval(['load mt' num2str(y) ';']);
eval(['year=mt' num2str(y) ';']);
% determine year is leap year
if length(year)==8784
  A=1;
  year=zeros(8784,6);
else if length(year)==8760
  A=0;

Appendix VII/ VII
year1=zeros(8760,6);
else
A=FALSE;
end
end

% reorganize year for simple building model input
year1=year(:,2)/10,year(:,6),year1(:,3:6));

% cut off year to be normal length (necessary for function Irradf)
year1=year1(1:8760,:);
soldirn=year(:,3);
soldif0=year(:,1);

% calculating diffuse solar irradiation on vertical surface and
% direct solar irradiance on North, East, South and West facades.
% % North wall:
% [soldirN,soldifV]=irradf(soldirn,soldif0,elevat,cosincN,iday,surfpos(1,:),albedo);
% year1(:,3)=soldifV+soldirN;
% % East wall:
% [soldirE,soldifE]=irradf(soldirn,soldif0,elevat,cosincE,iday,surfpos(2,:),albedo);
% year1(:,4)=soldifV+soldirE;
% % South wall:
% [soldirS,soldifS]=irradf(soldirn,soldif0,elevat,cosincS,iday,surfpos(3,:),albedo);
% year1(:,5)=soldifV+soldirS;
% West wall:
[soldirW,soldifW]=irradf(soldirn,soldif0,elevat,cosincW,iday,surfpos(4,:),albedo);
year1(:,6)=soldifV+soldirW;

%reconstruct length of leap years
if A==1
    year1=[year1; year1(8737:8760,:)];
else if A==0
    year1=year1;
else
    A=FALSE;
end
end

eval(['mtSimp' num2str(y) '=year1;']);
eval(['save mtSimp' num2str(y) ' mtSimp' num2str(y)]);
end
VII.2 Inverse modeling process - lumped RC building model (Δt = 3600 s)

Inverse modeling process - set input and output (from virtual case study building) and parameter search space (1/4)

clear all;

global tu t lb ub FunHandle u u2 y Ti_VABI Tf1_VABI Tenv_VABI Q_solar_VABI % Pi

load VABI_results_processed

%% System identification with Climate Data NEN5060E
data = results_VABI_case_variant1_NEN5060E;
% data = results_VABI_case_variant2_NEN5060E;
% data = results_VABI_case_variant3_NEN5060E;
% data = results_VABI_case_variant4_NEN5060E;
% data = results_VABI_case_variant5_NEN5060E;

%% Validation with Climate Data NEN5060refTO5
data = results_VABI_case_variant1_NEN5060refTO5;
% data = results_VABI_case_variant2_NEN5060refTO5;
% data = results_VABI_case_variant3_NEN5060refTO5;
% data = results_VABI_case_variant4_NEN5060refTO5;
% data = results_VABI_case_variant5_NEN5060refTO5;

%% Reference data Climate Data NEN 5060 Energy
Irrad = NEN5060E(:,6);
GHI = room1(:,1);
Te = room1(:,2);
Toi_VABI = room1(:,3);
Ti_VABI = room1(:,4);
Tisupply_VABI = room1(:,5);
qmvent = room1(:,6) + room1(:,8);
P = room1(:,7) + room1(:,9);
Q_solar_VABI = room1(:,10);
Q_int_VABI = room1(:,11);
Tenv = room1(:,16:20);
Tenv_VABI = mean(Tenv,2);
Tfl_VABI = room1(:,21);
% Floorsystwsup = room1(:,31);
% Floorsystwret = room1(:,32);

%% Reference data Climate Data NEN5060refTO5
Irrad2 = NEN5060refTO5(:,6);
GHI2 = room2(:,1);
Te2 = room2(:,2);
Toi_VABI2 = room2(:,3);
Ti_VABI2 = room2(:,4);
Tisupply_VABI2 = room2(:,5);
qmvent2 = room2(:,6) + room2(:,8);
P2 = room2(:,7) + room2(:,9);
Q_solar_VABI2 = room2(:,10);
Q_int_VABI2 = room2(:,11);
Tenv2 = room2(:,16:20);
Tenv_VABI2 = mean(Tenv2,2);
Tfl_VABI2 = room2(:,21);
% Floorsystwsup2 = room2(:,31);
% Floorsystwret2 = room2(:,32);

%% specification u
u=[Te Irrad Q_int_VABI P];
u2=[Te2 Irrad2 Q_int_VABI2 P2];

L=length(u);
nc1Data=size(y);
nt=nc1Data(1); % Number of time steps
dt=3600; % timestep [s]
tu=0:dt:nt-1;dt;
t=1:nt;
FunHandle=@RMSE_optTi;

%% parameters to be identified:
% p(1) = Genv,i [W/K]
% p(2) = Gfast [W/K]
% p(3) = Gint [W/K]
%% expected parameter values:
% p(1) = Genv,i [W/K]
% p(2) = Gfast [W/K]
% p(3) = Gint [W/K]
% p(4) = Gfl,conv [W/K]
% p(5) = Cenv [J/K]
% p(6) = Ci [J/K]
% p(7) = Cint [J/K]
% p(8) = Cfl [J/K]
% p(9) = Awin [m²]

%% lower and upper bounds
lb = 0.1*i;
ub = 10*i;

Inverse modeling process - start three-stage inverse modeling approach (2/4)

initialisation_modelT % call initial model (input and output variables, parameter ranges)
NVARS=9; % number of parameters which need to be identified.
gaopts = gaoptimset('CreationFcn',@gacreationlinearfeasible,'PopulationSize',
15*NVARS,'EliteCount',4,'CrossoverFraction',0.70,'MutationFcn],[mutationadaptfeasible,1,0.60],'PlotFcns',
[@gaplotbestf @gaplotdistance @gaplotrange],'StallGenLimit',50);
psopts = psoptimset('PlotFcns',@psplotbestf,'MaxIter', 100*NVARS,'TolFun',le-6,'MaxFunEvals', 2000*NVARS);
fopts=optimset('Algorithm', 'interior-point');

for n=[30] % for reproducibility; define multiple Genetic Algorithm generations
opts =gaoptimset(gaopts,'Generations',n*NVARS); % options Genetic Algorithm
[X,FVAL,EXITFLAG,OUTPUT] = ga(FunHandle,NVARS,[],[],[],[],lb,ub,[],opts); % run Genetic Algorithm
SsqFunSS_optT_sim % results for solution Genetic Algorithm (NEN5060refTO5)
[Xp, FVALp, EXITFLAGp,OUTPUTp] = patternsearch(FunHandle,X,[],[],[],[],lb,ub,psopts); % run PatternSearch
SsqFunSS_optT2_sim % results for solution GA + PatternSearch (NEN5060refTO5)
[Ti_mod2 = Ti_mod2';
Tfl_mod2 = Tfl_mod2';
SsqFunSS_optT4_sim % results for solution GA + PatternSearch + Fmincon (NEN5060E)
Ti_mod = Ti_mod';
Tfl_mod = Tfl_mod';
record = [record; n FVAL MBE RMSE G X FVALp MBEp RMSEp Gp Xf Ti_mod2 Tfl_mod2 Ti_mod Tfl_mod]
end

n = record(:,1);
FVAL = record(:,2);
MBE = record(:,3);
RMSE = record(:,4);
G = record(:,5);
X = record(:,6:14);
FVALp = record(:,15);
MBEp = record(:,16);
RMSEp = record(:,17);
Gp = record(:,18);
Xp = record(:,19:27);
FVALf = record(:,28);
MBEf = record(:,29);
RMSEf = record(:,30);
Gf = record(:,31);
Xf = record(:,32:40);
Tf1_mod2 = record(:,41:8800);
Tf1_mod = record(:,8801:17560);
T1_mod = record(:,17561:26320);
Tfl_mod = record(:,26321:35080);
Inverse modeling process - calculate objective function value for each combination of parameters defined by algorithm (3/4)

```matlab
% clear all;

function q=RMSE_optTi(p)

global Ti_VABI Tfl_VABI Tenv_VABI tu u

%% parameters
% (1) = p(1)/p(5); % Genv,i/Cenv         [s^-1]
% (2) = p(1)/p(6); % Genv,i/Ci           [s^-1]
% (3) = p(2)/p(6); % Gfast/Ci            [s^-1]
% (4) = p(3)/p(6); % Gint/Ci             [s^-1]
% (5) = p(4)/p(6); % Gfl,conv/Ci         [s^-1]
% (6) = p(3)/p(7); % Gint/Cint           [s^-1]
% (7) = p(4)/p(8); % Gfl,conv/Cfl        [s^-1]
% (8) = p(9)/p(7); % Awin/Cint           [m2K/J]

%% construct state space matrices:
% [ Tenv              Ti                                         Tint            Tfl          ]
A = [(-(p(1)/p(5))) (p(1)/p(5))                                     0              0              ; % Cenv
     (p(1)/p(6))  (-(p(1)/p(6))-(p(2)/p(6))-(p(3)/p(6))-(p(4)/p(6)) (p(3)/p(6)     (p(4)/p(6))    ; % Ci
     0              (p(3)/p(7))                                     (-(p(3)/p(7))) 0              ; % Cint
     0              (p(4)/p(8))                                     0              (-(p(4)/p(8)))]; % Cfl

% [ Te                IrradW            Qint           Power          ]
B = [0            0               0         0         ;  % Cenv
     (p(2)/p(6))  0               (1/p(6))  0         ;  % Ci
     0            (p(9)/p(7))     0         0         ;  % Cint
     0            0               0         (1/p(8)) ]; % Cfl

% [ Tenv  Ti  Tint  Tfl  ]
C= [0 1 0 0 ];

% Construct state space model structure:
sys=ss(A,B,C,D);

% initialize and simulate:
x0=[Tenv_VABI(1,1); Ti_VABI(1,1); Ti_VABI(1,1); Tfl_VABI(1,1)];
output=lsim(sys,u,tu,x0);

Ti_mod=output(:,1);

tt=10:8760;
error = Ti_mod(tt,1) - Ti_VABI(tt,1);
MSE=mse(error);
a=find((isnan(error))==1);
if a>0
    sseT=1e12;
else
    sseT= sqrt(MSE);   % RMSE
end
q=sseT;

Inverse modeling process - estimate T\textsubscript{e} and T\textsubscript{fl} for identified combination of parameters (4/4)

%% parameters
% (1) = Xf(1)/Xf(5); % Genv,i/Cenv         [s^-1]
% (2) = Xf(1)/Xf(6); % Genv,i/Ci           [s^-1]
% (3) = Xf(2)/Xf(6); % Gfast/Ci            [s^-1]
% (4) = Xf(3)/Xf(6); % Gint/Ci             [s^-1]
% (5) = Xf(4)/Xf(6); % Gfl,conv/Ci         [s^-1]
% (6) = Xf(3)/Xf(7); % Gint/Cint           [s^-1]
% (7) = Xf(4)/Xf(8); % Gfl,conv/Cfl        [s^-1]
% (8) = Xf(9)/Xf(7); % Awin/Cint           [m2K/J]
```
% state space matrices:
% [Tenv  Ti  Tint  Tf1   ]
A = [(-(Xf(1)/Xf(5)))  (Xf(1)/Xf(5))  0  0   ];  % Cenv 
    ([Xf(1)/Xf(6)]  (- (Xf(1)/Xf(6)) - (Xf(2)/Xf(6)) - (Xf(3)/Xf(6)) - (Xf(4)/Xf(6)))
    (Xf(3)/Xf(6))  (Xf(4)/Xf(6))  0  0   ];  % Ci
    0  (Xf(3)/Xf(7))     (- (Xf(3)/Xf(7)))    0   ];  % Cint
    0  (Xf(4)/Xf(8))     0     (- (Xf(4)/Xf(8))) ];  % Cfl

% [Te   Irrad  Qint  Power]
B = [0             0               0         0          ];  % Cenv 
    ([Xf(2)/Xf(6)]   0               (1/Xf(6)) 0          );  % Ci
    0             (Xf(9)/Xf(7))   0         0          );  % Cint
    0             0               0         (1/Xf(8))   ];  % Cfl

% [Tenv  Ti  Tint  Tf1]
C = [0 1 0 0   ];
    0 0 0 1 ];

% Construct state space model structure:
sys=ss(A,B,C,D);

% initialize and simulate:
x0=[Tenv_VABI2(1,1); Ti_VABI2(1,1); Ti_VABI2(1,1); Tfl_VABI2(1,1)];
output=lsim(sys,u2,tu,x0);
Ti_mod2=output(:,1);
Tfl_mod2=output(:,2);

tt=10:8760;
errorf = Ti_mod2(tt,1)-Ti_VABI2(tt,1);

% mean bias error:
k=length(tt);
MBEf = sum(errorf)/k;  % MBE_Ti over 8751 values (in °C)

% root mean squared error:
RMSEf=sqrt(MSEf);   % RMSE_Ti over 8751 values (in °C)

% Goodness of fit:
Gf =100*(1-norm(Ti_VABI2(tt,1)-Ti_mod2(tt,1))/norm(Ti_VABI2(tt,1)-mean(Ti_VABI2(tt,1)))); % G_Ti (in %)

Appendix VII / VII
VII.3 Determine $G_{pipes}$ - lumped RC building model ($\Delta t = 3600$ s)

```matlab
%% required data:
load VABI_results_processed;
% load setpoints;
load lumpedRCbuildingmodel_case_variant1;
% load lumpedRCbuildingmodel_case_variant2;
% load lumpedRCbuildingmodel_case_variant3;
% load lumpedRCbuildingmodel_case_variant4;
% load lumpedRCbuildingmodel_case_variant5;

record = results_identification;
n = record(:,1);
FVAL = record(:,2)/8751;
MBE = record(:,3);
RMSE = record(:,4);
G = record(:,5);
X = record(:,6:14);
FVALp = record(:,15)/8751;
MBEp = record(:,16);
RMSEp = record(:,17);
Gp = record(:,18);
Xp = record(:,19:27);
FVALf = record(:,28)/8751;
MBEf = record(:,29);
RMSEf = record(:,30);
Gf = record(:,31);
Xf = record(:,32:40);
Ti_mod2 = record(:,41:8800);
Tfl_mod2 = record(:,8801:17560);
Ti_mod = record(:,17561:26320);
Tfl_mod = record(:,26321:35080);

data = results_VABI_case_variant1_NEN5060E;
% data = results_VABI_case_variant2_NEN5060E;
% data = results_VABI_case_variant3_NEN5060E;
% data = results_VABI_case_variant4_NEN5060E;
% data = results_VABI_case_variant5_NEN5060E;

% RC model:
Te = data(:,2);
% RC model:
Ti_RC = Ti_mod';
Tfl_RC = Tfl_mod';

% VABI model:
Twret_VABI = data(:,32);
Power_VABI = data(:,7) + data(:,9);

% Control settings:
Theat_set = setpoints(:,1);
Tcool_set = setpoints(:,2);

for o=1:8760
    if Te < 12.1
        Twsup(o,1) = 34.09-1.0909*Te(o,1);
    else
        Twsup(o,1) = 20.028-0.1676*Te(o,1);
    end
end

%% determine $G_{pipes}$:
for i=2:8760
    dTheat(i,1) = Theat_set(i,1) - Theat_set(i-1,1);
dTcool(i,1) = Tcool_set(i,1) - Tcool_set(i-1,1);
end

for n = 1:8760
    if (Power_VABI(n,1) > 0 & dTheat(n,1) == 2) | (Power_VABI(n,1) < 0 & dTcool(n,1) == -2)
        c(n,i) = (Twret_VABI(n,1) - Tfl_RC(n,i))/(Twsup(n,1) - Tfl_RC(n,i));
        % calculate $c = e^{-(G_{pipes}/m,w*c,pw)}$ if VABI power is activated during entire time period 3600 s
        % when set point has increased in case of heating or set point has decreased in case of cooling
    else
        c(n,i) = 0;
    end
end
```
for i=1:100  % for reproducibility; define which inverse modeling solution
    for n = 1:8760
        if (c(n,i) > 0 && c(n,i) < 1.0) % c should be a constant between 0 and 1
            c_num(n,i) = 1;
            c_real(n,i) = c(n,i);
        else
            c_num(n,i) = 0;
            c_real(n,i) = 0;
        end
    end
end

c_size = sum(c_num);
c_sum = sum(c_real);
for i=1:100
    c_mean(i,1) = c_sum(1,i) / c_size(1,i);
end
Gpipes = -1* log(c_mean)*698;

%% Save results
load lumpedRCbuildingmodel_case_variant1_Gpipes;
% load lumpedRCbuildingmodel_case_variant2_Gpipes;
% load lumpedRCbuildingmodel_case_variant3_Gpipes;
% load lumpedRCbuildingmodel_case_variant4_Gpipes;
% load lumpedRCbuildingmodel_case_variant5_Gpipes;
VII.4 Estimate indoor air temperature and heating and cooling demands – coupled simulation model ($\Delta t = 60$ s)

Coupled simulation model – entire coupled simulation model (1/3)

Figure VII.6: Coupled simulation model in Simulink

Appendix VII/ VII
Coupled simulation model – control block part (entire coupled simulation model) (2/3)

Figure VII.7: Control of the coupled simulation model in Simulink

Coupled simulation model – Set input and lumped RC building model; start estimation with Simulink model (3/3)

```matlab
%% Load system identification results
load lumpedRCbuildingmodel_case_variant1_Gpipes;
% load lumpedRCbuildingmodel_case_variant2_Gpipes;
% load lumpedRCbuildingmodel_case_variant3_Gpipes;
% load lumpedRCbuildingmodel_case_variant4_Gpipes;
% load lumpedRCbuildingmodel_case_variant5_Gpipes;

%% Assign data = VABI results NEN5060E
data = results_VABI_case_variant1_NEN5060E;
% data = results_VABI_case_variant2_NEN5060E;
% data = results_VABI_case_variant3_NEN5060E;
% data = results_VABI_case_variant4_NEN5060E;
% data = results_VABI_case_variant5_NEN5060E;

%% Assign data = VABI results NEN5060refT05
data = results_VABI_case_variant1_NEN5060refT05;
% data = results_VABI_case_variant2_NEN5060refT05;
% data = results_VABI_case_variant3_NEN5060refT05;
% data = results_VABI_case_variant4_NEN5060refT05;
% data = results_VABI_case_variant5_NEN5060refT05;

%% Define input parameters and control settings
% NEN 5060E data:
t = 3600*(0:8759)';
usimulink = [t NEN5060E(:,1) NEN5060E(:,6) data(:,11)];
```
for o=1:8760  
if u_simulink(o,2) < 12.1  
  Twsup(o,1) = 34.09-1.0909*u_simulink(o,2);  
  heatingOnOff(o,1) = 1;  
  if (Ti_mod < Ti_heat and Te < 12.1)  
    heatingOnOff(o,1) = 1;  
  end  
else  
  Twsup(o,1) = 20.028-0.1676*u_simulink(o,2);  
  heatingOnOff(o,1) = 0;  
  if (Ti_mod > Ti_cool and Te > 12.1)  
    heatingOnOff(o,1) = 1;  
  end  
end  
heatingOnOff = [t heatingOnOff];  
coolingOnOff = [t coolingOnOff];  
Twsup = [t Twsup];  
l = ones(8760);  
l = l(:,1);  

%% Loop to simulate heating and cooling demand with Simulink model  
clear record A B C D sys  
record = [];  
for i = 1;  
  Xf = Xf(i,:);  
  X = Xf(i,:);  
  % identified lumped RC building model parameters  
  % construct state space matrices:  
  % [ Tenv  Ti  Tint  Tfl ]  
  A = [ (-(X(1)/X(5))) (X(1)/X(5)) 0 0;  
        (X(2)/X(6)) (-(X(1)/X(6)))-(X(2)/X(6))-(X(3)/X(6))-(X(4)/X(6));  
        0 (X(3)/X(7)) 0;  
        0 (X(4)/X(8)) 0; ];  
  % [Te  IrradW  Qint  Power]  
  B = [0 0 0 0;  
       (X(2)/X(6)) 0 (1/X(6)) 0;  
       0 (X(9)/X(7)) 0;  
       0 0 (1/X(8)); ];  
  % [Tenv  Ti  Tint  Tfl ]  
  C = [0 0 0 0;  
       1 0 0 0;  
       0 1 0 0;  
       0 0 1 0; ];  
  % [ Te  IrradW  Qint  Twmean]  
  D = [1 0 0 0;  
       0 0 0 0;  
       0 0 0 0;  
       0 0 0 0; ];  

  SimOut = sim('heating_curve_model_fixed_c');  
  run Compare_with_Vabi;  
  ymod = ymod';  
  Ti_Simulink = Ti_Simulink';  
  Tfl_Simulink = Tfl_Simulink';  
  record = [record; i FVALf(i,:) MBEf(i,:) RMSEf(i,:) Gf(i,:) X Ti_mod(i,:) Tfl_mod(i,:) MBEe RMSEe Ge MBEt RMSEt Gt APEheat APEcool APEtot ymod(1,1:8760) Ti_Simulink(1,1:8760) Tfl_Simulink(1,1:8760)];  
end  
n = record(:,1);  
FVALf = record(:,2)/8751;  
MBEf = record(:,3);  
RMSEf = record(:,4);  
Gf = record(:,5);  
Xf = record(:,6:14);  
Ti_RC = record(:,15:8774);  
Tfl_RC = record(:,8775:17534);  
MBEe = record(:,17535);  
RMSEe = record(:,17536);  
Ge = record(:,17537);  
MBEt = record(:,17538);  
RMSEt = record(:,17539);  
Gt = record(:,17540);  
APEheat = record(:,17541);  
APEcool = record(:,17542);
APEtot = record(:,17543);
ymod = record(:,17544:26303);
TfI_Simulink = record(:,26304:35063);
Tfl_Simulink = record(:,35064:43823);

%%% Save results NEN5060E
data = results_coupledSimulationmodel_case_variant1_NEN5060E;
data = results_coupledSimulationmodel_case_variant2_NEN5060E;
data = results_coupledSimulationmodel_case_variant3_NEN5060E;
data = results_coupledSimulationmodel_case_variant4_NEN5060E;
data = results_coupledSimulationmodel_case_variant5_NEN5060E;

%%% Save results NEN5060refTO5
data = results_coupledSimulationmodel_case_variant1_NEN5060refTO5;
data = results_coupledSimulationmodel_case_variant2_NEN5060refTO5;
data = results_coupledSimulationmodel_case_variant3_NEN5060refTO5;
data = results_coupledSimulationmodel_case_variant4_NEN5060refTO5;
data = results_coupledSimulationmodel_case_variant5_NEN5060refTO5;
VII.5 Data processing

Data processing - determine output with $\Delta t = 3600$ s (1/2)

for $n=1:525600$

if output60(o,13) < output60(o,6) & output60(o,1) < 12.1 & output60(o,13) > output60(o,5) & output60(o,1) > 12; 
output60(o,17) = 1; 
output60(o,18) = output60(o,16) * (1/60); 
else 
output60(o,17) = 0; 
output60(o,18) = 0; 
end
end

Heating = output60(:,18); % Heating if power > 0 ($\Delta t = 60s$)
Heating(Heating<0) = 0;
Cooling = output60(:,18); % Cooling if power < 0 ($\Delta t = 60s$)
Cooling(Cooling>0) = 0;

for $n=1:8760$

Heatingmod(n,1) = sum(Heatingmod((n-1)*24+1:n*24,1)); % Heating_power_mod $\Delta t = 3600$s
Coolingmod(n,1) = sum(Coolingmod((n-1)*24+1:n*24,1)); % Cooling_power_mod $\Delta t = 3600$s
end

P_mod = Heatingmod + Coolingmod; % Power_mod $\Delta t = 3600$s
Tmod = output3600; % Estimated temperatures $\Delta t = 3600$s
P = data(:,7) + data(:,9); % Power_VABI $\Delta t = 3600$s
Ti_Simulink = Tmod(:,13); % Ti_mod $\Delta t = 3600$s
Tfl_Simulink = Tmod(:,15); % Tfl_mod $\Delta t = 3600$s

% [HeatingRC HeatingVABI CoolingRC CoolingVABI]
for $n=1:365$

Eday(n,1) = sum(HeatingVABI((n-1)*24+1:n*24,1)); % E_heat_day_VABI
Eday(n,3) = sum(CoolingVABI((n-1)*24+1:n*24,1)); % E_cool_day_VABI
end

for $n=1:52$

Eweek(n,1) = sum(HeatingVABI((n-1)*168+1:n*168,1)); % E_heat_week_VABI
Eweek(n,3) = sum(CoolingVABI((n-1)*168+1:n*168,1)); % E_cool_week_VABI
end

for $n=1:12$

Emonth(n,1) = sum(HeatingVABI((n-1)*730+1:n*730,1)); % E_heat_month_VABI
Emonth(n,3) = sum(CoolingVABI((n-1)*730+1:n*730,1)); % E_cool_month_VABI
end

Eyear(1,1) = sum(HeatingVABI(:,1)); % E_heat_year_VABI
Eyear(1,3) = -abs(sum(CoolingVABI(:,1))); % E_cool_year_VABI
Eyear = [Eyear;Eyear];
tt=10:8760;
error1 = P_mod(tt,1)-P(tt,1); % residual P
% mean bias error:
k=length(tt);
MBEe = sum(error1)/k; % MBE_power over 8751 values (in W)

% root mean squared error:
MSEe=mse(error1);
RMSEe=sqrt(MSEe); % RMSE_power over 8751 values (in W)

% normalized root mean squared error:
NRMSEe = RMSEe / (max(HeatingVABI(:,1)) % NRMSE_power over 8751 values (in %)

% Goodness of fit:
Ge =100*(1-norm(y(tt,1)-ymod(tt,1))/norm(y(tt,1)-mean(y(tt,1)))); % G_power (in %)

errort = Ti_Simulink(tt,1)-data(tt,3); % residual Ti

% mean bias error:
k=length(tt);
MBEt = sum(errort)/k; % MBE_Ti over 8751 values (in °C)

% root mean squared error:
MSEt=mse(errort);
RMSEt=sqrt(MSEt); % RMSE_Ti over 8751 values (in °C)

% Goodness of fit:
Gt =100*(1-norm(data(tt,3)-Ti_Simulink(tt,1))/norm(data(tt,3)-mean(data(tt,3)))); % G_Ti (in %)

% annual demands
HeatingVABItot = sum(HeatingVABI); % emulated annual heating demand
CoolingVABItot = sum(CoolingVABI); % emulated annual cooling demand
Heatingmodtot = sum(Heatingmod); % estimated annual heating demand
Coolingmodtot = sum(Coolingmod); % estimated annual cooling demand

% Percentage errors annual demands
PEheat = ((HeatingVABItot - Heatingmodtot)/HeatingVABItot)*100; % PE_heat_a (in %)
PEcool = ((CoolingVABItot - Coolingmodtot)/CoolingVABItot)*100; % PE_cool_a (in %)

Data processing - cross-correlation test (2/2)

%% cross-correlation function matlab
measdata = iddata(Ti_VABI(25:8736,1),u(25:8736,:),3600);
sys1 = idss(sys)
figure (5)
resid(sys1,measdata,'Corr',24);

%% define cross-correlation manually

% Residual Ti
errorf = Ti_mod(tt,1)-Ti_VABI(tt,1);

% sum residuals Ti
errortot = sum(errorf);

% input means
Temean = sum(Te(25:8736,1))/8712;
Irradmean = sum(Irrad(25:8736,1))/8712;
Qintmean = sum(Q_int_VABI(25:8736,1))/8712;
ymean = sum(y(25:8736,1))/8712;
residual_mean = sum(errorf(25:8736,1))/8712;

% input and residual variance at lag = 0
for t = 25:8736
    c_u1(t-24,1) = (Te(t,1)-Temean); % Te
    c_u2(t-24,1) = (Irrad(t,1)-Irradmean); % Irrad
    c_u3(t-24,1) = (Q_int_VABI(t,1)-Qintmean); % Qint
    c_u4(t-24,1) = (y(t,1)-ymean); % Power
end

for t = 25:8736
    for lag = -24:24
        c_y(t-24,lag+25) = (errorf(t+lag,1)-residual_mean); % Residual Ti
    end
end
for t = 25:8736
    c_u1sq(t-24,1) = (c_u1(t-24,1))^2;  % Te
    c_u2sq(t-24,1) = (c_u2(t-24,1))^2;  % Irrad
    c_u3sq(t-24,1) = (c_u3(t-24,1))^2;   % Qint
    c_u4sq(t-24,1) = (c_u4(t-24,1))^2;      % Power
    c_y_sq(t-24,1) = (c_y(t-24,25))^2;      % Residual Ti
end

c_uu1(1,1) = sum(c_u1sq(:,1));  % Te
    c_uu2(1,1) = sum(c_u2sq(:,1));% Irrad
    c_uu3(1,1) = sum(c_u3sq(:,1));   % Qint
    c_uu4(1,1) = sum(c_u4sq(:,1));   % Power
    c_yy(1,1) = sum(c_y_sq(:,1));       % Residual Ti

% sample cross-covariance function for lags -24 till +24
for t = 25:8736
    for lag = -24:24
        c_u1y(t-24,lag+25) = c_u1(t-24,1) * c_y(t-24,lag+25);     % Te
        c_u2y(t-24,lag+25) = c_u2(t-24,1) * c_y(t-24,lag+25);     % Irrad
        c_u3y(t-24,lag+25) = c_u3(t-24,1) * c_y(t-24,lag+25);     % Qint
        c_u4y(t-24,lag+25) = c_u4(t-24,1) * c_y(t-24,lag+25);      % Power
    end
end

for lag = -24:24
    c_u1y_tot(lag+25,1) = sum(c_u1y(:,lag+25));  % Te
    c_u2y_tot(lag+25,1) = sum(c_u2y(:,lag+25));     % Irrad
    c_u3y_tot(lag+25,1) = sum(c_u3y(:,lag+25));     % Qint
    c_u4y_tot(lag+25,1) = sum(c_u4y(:,lag+25));      % Power
end

% cross-correlation coefficient r for lags -24 till +24
for lag = -24:24
    r_u1y(lag+25,1) = (c_u1y_tot(lag+25,1)/8712) / ( sqrt((c_uu1/8712) * (c_yy/8712)) );   % Te
    r_u2y(lag+25,1) = (c_u2y_tot(lag+25,1)/8712) / ( sqrt((c_uu2/8712) * (c_yy/8712)) );     % Irrad
    r_u3y(lag+25,1) = (c_u3y_tot(lag+25,1)/8712) / ( sqrt((c_uu3/8712) * (c_yy/8712)) );     % Qint
    r_u4y(lag+25,1) = (c_u4y_tot(lag+25,1)/8712) / ( sqrt((c_uu4/8712) * (c_yy/8712)) );      % Power
end

% cross-correlation coefficient r for lags -24 untill +24 (also possible)
% for lag = -24:24
%    r_u1y(lag+25,2) = ( (c_u1y_tot(lag+25,1)/8712) / ( sqrt((c_uu1/8712) * (c_yy/8712)) ));  % Te
%    r_u2y(lag+25,2) = ( (c_u2y_tot(lag+25,1)/8712) / ( sqrt((c_uu2/8712) * (c_yy/8712)) )); % Irrad
%    r_u3y(lag+25,2) = ( (c_u3y_tot(lag+25,1)/8712) / ( sqrt((c_uu3/8712) * (c_yy/8712)) ));  % Qint
%    r_u4y(lag+25,2) = ( (c_u4y_tot(lag+25,1)/8712) / ( sqrt((c_uu4/8712) * (c_yy/8712)) ));  % Power
% end

% plot cross-correlation coefficient r as function of lags -24 till +24
% cross-correlation coefficients r for input Te
lag = -24:24;
figure(1);
subplot(2,2,1);
plot(lag,r_u1y(:,1)*-1,'.k');
set(gca,'XLim',[-25 25], 'YLim',[-0.41 0.41]);
grid on;
grid minor;
ylabel('r');
xlabel('lag');

% cross-correlation coefficients r for input Irrad
subplot(2,2,2);
plot(lag,r_u2y(:,1)*-1,'.k');
set(gca,'XLim',[-25 25], 'YLim',[-0.41 0.41]);
grid on;
grid minor;
ylabel('r');
xlabel('lag');

% cross-correlation coefficients r for input Qint
subplot(2,2,3);
plot(lag,r_u3y(:,1)*-1,'.k');
set(gca,'XLim',[-25 25], 'YLim',[-0.41 0.41]);
grid on;
grid minor;
ylabel('r');
xlabel('lag');

% cross-correlation coefficients r for input Power
subplot(2,2,4);
plot(lag,r_u4y(:,1)*-1,'.k');
set(gca,'XLim',[-25 25], 'YLim',[-0.41 0.41]);
grid on;
grid minor;
ylabel('r');
xlabel('lag');

Appendix VII/ VII
% cross-correlation coefficients r for input P

subplot(2,2,4);
plot(lag, r_u4y(:,1)*-1, '.k');
set(gca, 'XLim', [-25 25], 'YLim', [-0.41 0.41]);
grid on;
grid minor;
ylabel('r');
xlabel('lag');