Common exercise on data analysis, opaque wall, methodologies and preliminary results

Citation for published version (APA):

Document status and date:
Published: 01/01/2012

Document Version:
Accepted manuscript including changes made at the peer-review stage

Please check the document version of this publication:
• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.
Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.

Download date: 27. Dec. 2019
ST3 Common Exercise on Data Analysis

Opaque Wall

Methodologies and Preliminary Results

Jos van Schijndel

Eindhoven University of Technology

February 2012
# Introduction ................................................................................................................. 3

1. Data analysis ............................................................................................................. 4
   1.1 Remarks in advance ........................................................................................... 4
      1.1.1 Definition of air to air U value ................................................................. 4
      1.1.2 Missing parameters .................................................................................. 4
      1.1.3 Location of the sensors ............................................................................ 5
   1.2 Estimation of the U value .................................................................................. 5
      1.2.1 Moving average wall resistance ............................................................... 5
      1.2.2 Daily fluctuation ..................................................................................... 6

2. Grey-Box modeling using ODEs and MatLab ...................................................... 8
   2.1 Designing the model ....................................................................................... 8
   2.2 Implementing of the ODEs model into MatLab ............................................. 8
   2.3 Calibrating the variable parameters R and C ............................................. 9

3. Black-Box modeling using MatLab Identification Toolbox ......................... 10

4. Discussion ............................................................................................................. 12

5. Appendix A, MatLab Grey-Box mfiles ............................................................. 13
Introduction

The objective of this common exercise is to identify strengths and weaknesses (reliability, inherent physical information, ...) of different available models. The proposed case study consists on a very simple lightweight opaque and homogeneous wall. This test component was chosen simple and the tests were intentionally oversized regarding measured quantities, test period length, and test conditions, in order to investigate capabilities and limitations of different models and methods. The simplicity in the component and the oversize in the data provide enough freedom to allow the application of a wide variety of different analysis and validation approaches, with different degree of complexity and accuracy.

The requested output is estimate the air to air U value from the given measurements.

This report presents the following two methodologies including preliminary results:

I. Data Analysis – The requested U value is estimated directly from measurements.

II. Grey-Box modeling using ODEs and MatLab – An ODEs based model of the construction is simulated using MatLab and its model parameters calibrated using the measurements.

III. Black-Box modeling using MatLab – A black box model is constructed for simulating the surface temperatures and the internal surface heat flux from indoor and outdoor air temperature and radiation.

The outline of report is as follows: Preliminary results of each of the four methodologies are provided in separate Sections. Additional information can be found in the Appendices.
1. Data analysis

1.1 Remarks in advance

1.1.1 Definition of air to air U value

The air to air U value is by definition dependent on the surface coefficients of heat transfer \( (h_i \text{ and } h_e) \). \( h_i \) and \( h_e \) are not constant over time. Furthermore the surface to surface U value of the wall can also vary over time for example due to rain. This means that the estimation of the U value is dependent on the time scale.

1.1.2 Missing parameters

Concerning the radiation details of the absorptance \( (a) \) and emittance \( (e) \) seems to be missing. These parameters could be estimated from the measurements but it is unclear whether this should be done or not.

Figure 1 shows measurements from March 12-16.

![Figure 1 Measurements from March 12-16.](image)

At day 14 it can be clearly seen that the external surface temperature \( (T_{se}) \) is lower than the external air temperature \( (T_{e}) \). The explanation could be net (long wave) radiation to the sky. The net radiation \( (L_{net}) \) seems to be missing. This quantity can be estimated using the vertical long wave radiation \( (G_{lw}) \), \( T_{se} \) and \( L_{net}=G_{lw}-5.67 \times 10^{-8} T_{se}^4 \) but is this accurate enough?
1.1.3 Location of the sensors

Placing sensors always disturbs the quantities to be measured. Details of the sensors’ locations, geometries and materials seem to be missing. Therefore it is very difficult to estimate the undisturbed situation.

1.2 Estimation of the U value

1.2.1 Moving average wall resistance

Figure 2 shows the moving average of several parameters from February till July:

![Graph of moving average parameters](image)

*Figure 2. Moving average of surface temperatures external (Tsecm), internal (Tsicm), internal heat flux (Qicm) and resistance (Rcm) from February till July.*

The moving average is calculated by:

\[ x_{cm}(i) = \text{mean}(x(1), \ldots, x(i)) \]

\[ R_{cm}(i) = \frac{T_{sicm}(i) - T_{secm}(i)}{Q_{icm}(i)} \]

- From day 0 – 20: \( R_{cm} \) is stabilizing;
- From day 20 – 45: \( R_{cm} = 0.5 \pm 0.02; \) \( U_{surface-surface} = 2 \pm 0.1 \)
- From day 45 – 130: \( R_{cm} \) is decreasing to 0.4;
- From day 130 - 150: \( R_{cm} = 0.4 \pm 0.02; \) \( U_{surface-surface} = 2.5 \pm 0.1 \)
1.2.2 Daily fluctuation

Figure 3 presents quantities from February till July during sunset:

![Figure 3](image)

Where, $U_{mat} = \frac{Q_i}{T_{si}-T_{se}}$

From Figure 3 top & bottom, from day 60 there seems to some structural change. From this day $T_{se}$ is significantly higher than $T_e$. Also the $U_{surface}$ to surface value is significantly higher than before.

Figure 4 shows the same parameters of figure 3 for March.
From figure 4 top, we focus on nights where all temperatures and Qi are stable. For these cases Umat (=U surface to surface) can be estimated from the bottom part. For night number 3 we have good conditions and we can estimate $2.2 < U_{mat} < 2.7$. 
2. Grey-Box modeling using ODEs and MatLab

2.1 Designing the model.

We designed a 3-State (3S) model as follows:

\[
\begin{align*}
C_1 \frac{dT_1}{dt} &= h_e(T_e(t) - T_1) - \frac{(T_1 - T_2)}{R_1} + a_I I(t) \\
C_2 \frac{dT_2}{dt} &= \frac{(T_1 - T_2)}{R_1} - \frac{(T_2 - T_3)}{R_2} \\
C_3 \frac{dT_3}{dt} &= h_i(T_i(t) - T_3) + \frac{(T_2 - T_3)}{R_2}
\end{align*}
\]

Where

Input:  
- \(T_e(t)\) external air temperature [°C] 
- \(T_i(t)\) internal air temperature [°C] 
- \(I(t)\) external solar irradiance [W/m²]

States:  
- \(T_1\) external surface temperature [°C] 
- \(T_2\) mid wall temperature [°C] 
- \(T_3\) internal surface temperature [°C]

Parameters:  
- \(R_i\) heat resistance [m²°C/W] 
- \(C_i\) heat capacity [J/°Cm²] 
- \(a_I\) solar absorption factor [-] 
- \(h\) heat transfer surface coefficient [W/m²°C]

2.2 Implementing of the ODEs model into MatLab

We implemented the model into the ‘ModelEq3S2P.m’ file (see Appendix A), to be used by the standard ODE functions of MatLab. Also a start file ‘Calc_start_3S2P.m’ (see Appendix A) was created to simulate the model with fixed parameters: \(h_i=10; h_e=20; a_I = 0.8\) and variable parameters \(R\) (total surface to surface resistance) and \(C\) (total heat capacity). To test the model we simulate a first guess for the variable parameters \(R\) and \(C\):

\[
\text{ModPar}(1)=0.20/1 \ (=R) \ \text{K/W} \\
\text{ModPar}(2)=2000*840*0.10 \ (=C) \ \text{J/K}
\]

and a time period of five days in August.
2.3 Calibrating the variable parameters R and C

After implementing and testing of the ODEs model, a function to be minimized was written in MatLab ‘Ssqfun3S2P.m’ (see Appendix A). The objective is to minimize to sum of the squares of measured minus simulated surface temperatures. Figure 5 shows the result:

![Graph showing the results of the optimization.](image)

At this point we are facing the problem whether to proceed yes or no with improving the simulated results. Possible improvements could be: (1) higher order model; (2) include long wave radiation; (3) extension of calibrated parameters for example hi, h_i, a_i. However, as already mentioned in Section 1, there are details missing causing uncertainties that are very difficult to estimate without knowing these details. Furthermore, from figure 5 bottom, the measured indoor surface temperature (Tsi) has some high frequency fluctuations (even higher than the indoor temperature fluctuations) that are not predict by the model. This could also be an indication that more details are missing.

We end this section with a question: ‘Should we model better or measure better?’
3. **Black-Box modeling using MatLab Identification Toolbox**

The MatLab identification toolbox was used to simulate Tse, Tsi and Qi as output with Te, Ti and Gv as input during February. Amongst other models, the linear parametric model ARX and State space were used. Figure 6 shows the result.

![Graph showing simulation results](image)

*Figure 6. Results of black modeling during February*

In order to calculate the U value, a step response of (Te = 0; Ti = 20) was simulated. Figure 7 shows the result.
The end values for the n4s4 model are: $T_{se} = 1.75$, $T_{si} = 16.49$, $Q_i = -32.4$ 
The end values for the arx221 model are: $T_{se} = 0.62$, $T_{si} = 16.12$, $Q_i = -36.8$

U air to air: $U_{n4s4} = 1.6$; $U_{arx221} = 1.8$;

Surface coefficients: $h_{i,n4s4} = 9.3$; $h_{i,arx221} = 9.4$; $h_{e,n4s4} = 18.5$; $h_{e,arx221} = 60$;

Heat resistance of the material (surface to surface): $R_{n4s4} = 0.45$; $R_{arx221} = 0.42$. 
4. Discussion

Summary of the results:

<table>
<thead>
<tr>
<th>Method</th>
<th>R surface to surface</th>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moving average direct from measured data</td>
<td>0.4 .. 0.5</td>
<td>1</td>
</tr>
<tr>
<td>Stable platforms direct from measured data</td>
<td>0.37 .. 0.45</td>
<td>1</td>
</tr>
<tr>
<td>Grey Box Modeling</td>
<td>0.2</td>
<td>2</td>
</tr>
<tr>
<td>Black Box Modeling</td>
<td>0.42 .. 0.45</td>
<td>3</td>
</tr>
</tbody>
</table>

Three methods provide quite close values for the heat resistance of the material (surface to surface).

The Grey Box Modeling approach still needs extra attention because it produces results that do not match the other methods. This is for ongoing research (see below).

The Black Box modeling approach arx221 provides the best result so far. Using all data of February, the estimated U air to air is 1.8;

**Ongoing research**

1. Round Robin Experiment Common Exercise 1, Design of the test Box Experiment. We use Comsol Multiphysics as a 3D reference model to simulate the output of virtual placed sensors located in the virtual Box experiment

2. A revisit of all methods presented in this report work but now using the simulated data from 1.
5. Appendix A, MatLab Grey-Box mfiles

%CALC_START_3S2P
%Start transfer model parameter optimisation, al, parameters at once
% JvS 2012/01

clear all
close all

global ClData ModPar Tse Tsi

A=importdata('August_2010.txt');
% 1:Day_J
% 2:Te
% 3:Tl
% 4:Tsi
% 5:Tse
% 6:Qi
% 7:Gv
% 8:Glw
% 9:WV
%10:WD
%11:RH

nt=length(A.data(:,1));
nt=5*1440;
tdag=A.data(1:nt,1)-A.data(1,1);
thour=24*tdag;
tsec=3600*thour;
Te=A.data(1:nt,2);
Ti=A.data(1:nt,3);
Tsi=A.data(1:nt,4);
Tse=A.data(1:nt,5);
Qi=A.data(1:nt,6);
Gv=A.data(1:nt,7);
Glw=A.data(1:nt,8);
WV=A.data(1:nt,9);
WD=A.data(1:nt,10);
RH=A.data(1:nt,11);

ClData=[Te Ti Gv Glw ];

nClData=size(ClData);
nt=nClData(1); % Number of time steps
dt=60; % timestep [s]
tu=0:dt:(nt-1)*dt;

% initial guess
ModPar(1)=0.20/1; % K/W
ModPar(2)=2000*840*0.10; % J/K
%Simulate First trial
x0=[Tse(1) Tse(1)/2+Tsi(1)/2  Tsi(1)'];
[t,x]=ode23tb('ModelEq3S2P',tu,x0);

%Interpolate for hourly steps
Tsim=interp1(t,x,tu);

%Plot 1
figure(1)
subplot(211)
plot(tu/60,Tse,'b',tu/60,Tsim(:,1),'r',tu/60,Te,'k')
ylabel('Tse')
legend('Tse meas','Tse sim','Te')

subplot(212)
plot(tu/60,Tsi,'b',tu/60,Tsim(:,3),'r',tu/60,Ti,'k')
ylabel('Tsi')
xlabel('Time [hr]')
legend('Tsi meas','Tsi sim','Ti')
drawnow

%Start Optimization using initial guess as input
est=fminsearch('Ssqfun3S2P',ModPar);

%Simulate optimal solution
ModPar=est;
% ModPar(1)=0.189;
% ModPar(2)=185620;

x0=[Tse(1) Tse(1)/2+Tsi(1)/2  Tsi(1)'];
[t,x]=ode23tb('ModelEq3S2P',tu,x0);

%Interpolate for hourly steps
Tsim=interp1(t,x,tu);

%Plot 1
figure(1)
subplot(211)
plot(tu/3600,Tse,'b',tu/3600,Tsim(:,1),'r',tu/3600,Te,'k')
title(['R = ', num2str(ModPar(1)),' K/W; C= ',num2str(ModPar(2)),'J/K'])
ylabel('Tse')
legend('Tse meas','Tse sim','Te')

subplot(212)
plot(tu/3600,Tsi,'b',tu/3600,Tsim(:,3),'r',tu/3600,Ti,'k')
ylabel('Tsi')
xlabel('Time [hr]')
legend('Tsi meas','Tsi sim','Ti')
drawnow
function xdot=modeleq3s2p(t,x)

global ClData ModPar

xdot=zeros(3,1);

Rtot=ModPar(1);   % K/W
Ctot=ModPar(2);   % J/K

R1=Rtot/2;
R2=Rtot/2;
C1=Ctot/4;
C2=Ctot/2;
C3=Ctot/4;

he=20;
hi=10;
aQ=0.8;

nClData=size(ClData);
nt=nClData(1);        % Number vof time steps
dt=60;                % timestep [s]
tu=0:dt:(nt-1)*dt;

% ClData=[Te RHe Irrad Ti RHi];
Tet=interp1(tu,ClData(:,1),t);
Tit=interp1(tu,ClData(:,2),t);
Qrt=interp1(tu,ClData(:,3),t);

xdot(1)=( he*(Tet -x(1))   - (x(1)-x(2))/R1   +aQ*Qrt          )/C1;
xdot(2)=(                    (x(1)-x(2))/R1 - (x(2)-x(3))/R2   )/C2;
xdot(3)=(-hi*(x(3)-Tit                      + (x(2)-x(3))/R2   )/C3;
function q=ssqfun3s2p(p)

global ClData ModPar Tse Tsi

nClData=size(ClData);
nt=nClData(1);          % Number vof time steps
dt=60;                % timestep [s]
tu=0:dt:(nt-1)*dt;

ModPar(1)=p(1);   % s
ModPar(2)=p(2);   % s

%Simulate First trial
x0=[Tse(1) Tse(1)/2+Tsi(1)/2  Tsi(1)]';
[t,x]=ode23tb('ModelEq3S2P',tu,x0);

%Interpolate for hourly steps
Tsim=interp1(t,x,tu);

%Plot 1
figure(1)
subplot(211)
plot(tu/60,Tse,'b',tu/60,Tsim(:,1),'r')
ylabel('Tse')
legend('Tse meas','Tse sim')
title(['R = ', num2str(p(1)),' K/W;  C= ',num2str(p(2)),'J/K '])
subplot(212)
plot(tu/60,Tsi,'b',tu/60,Tsim(:,3),'r')
ylabel('Tsi')
xlabel('Time [hr]')
legend('Tsi meas','Tsi sim')
drawnow

uTsi=Tsim(:,3)-Tsi;
uTse=Tsim(:,1)-Tse;

u=[uTsi;uTse];
%u=[uTsi];

q=sum(sum(u.^2));