

INTEGRATED BUILDING PHYSICS SIMULATION WITH FEMLAB/SIMULINK/MATLAB

A.W.M. (Jos) van Schijndel
University of Technology Eindhoven, Netherlands

ABSTRACT

The combined MatLab toolboxes FemLab and Simulink are evaluated as solvers for problems based on partial differential equations (PDEs). The FemLab software is designed to simulate systems of coupled PDEs, 1D, 2D or 3D, non-linear and time dependent. In order to show how the program works, a complete code for solving a 2D airflow problem is given. A validation study of this 2D dynamic airflow problem, modeled using Navier Stokes and buoyancy is shown. All results show good agreements with measurements. The following integrated building models are discussed: 1) A detailed case study on the combination of an indoor airflow model in FemLab connected to an on/off controller in Simulink. In this case, the code of a FemLab model is implemented in the discrete section of a S-Function of SimuLink, so that the specialized solvers of FemLab can be used. 2) Other already developed integrated building physics models in SimuLink are briefly discussed. These include a) a multi zone Heat Air and Moisture (HAM) indoor climate model, integration of this model with b) a thermal bridge model and controller and c) a moisture transport model and controller. At the conclusion the applications and the possible benefits for scientists, engineers and designers are discussed.

1. INTRODUCTION

Many scientific problems in building physics can be described by PDEs. There are a lot of software programs available in which one specific PDE is solved. They are developed in order to get the simulation results in a short time and are most often emphasized on the simplicity of input of data, e.g. geometrical data. A disadvantage is that they often are not very flexible when the user wants to change or combine models. Also they most often act as black boxes.

Another category of software like FemLab (COMSOL, 2000), is developed special for solving PDEs where the user in principle can simulate any

system of coupled PDEs (for example the Navier Stokes equations and combined heat and moisture transport). The user can focus on the model (PDE coefficients on the domain and boundary) and does not have to spend much time on solving and visualization. FemLab is a MatLab Toolbox (The Mathworks 1998) and has facilities to export models to Simulink (The Mathworks 1999) (Please note that the given references on Matlab, FemLab, and SimuLink are far from complete. They are intended as a starting point for further reading). The Simulink simulation environment is flexible so that already developed building and HVAC models, like for example SIMBAD (CSTB, 2001) can be coupled with the FemLab models. This combination can be very interesting for the following application areas:

- * Science. Scientists can fully concentrate on the physics behind the models and the validation of models.
- * Engineering. Engineers can combine or adapt already developed models for solving typical engineering problems using Simulink. Furthermore, it is easy to create Graphical User Interfaces (GUI) of the input of Simulink models, so engineers can develop user-friendly applications for end-users.
- * Design. Designers only have to concentrate on the parameters of the models and the outcome of the simulation results when using user-friendly developed applications in Simulink.

The further contents of the paper is described now. Section 2 shows how FemLab works, including some background information. Section 3 shows a complete example of an airflow problem, including the FemLab code and validation results. In Section 4, a detailed case study on the integration of the presented FemLab airflow model in SimuLink is discussed. Section 5 summarizes other developed integrated building physics models.

2. HOW FEMLAB WORKS

FemLab (COMSOL, 2000) is a toolbox written in MatLab (Mathworks, 1998). It solves systems of

coupled PDEs (up to 32 independent variables). The specified PDEs may be non-linear and time dependent and act on a 1D, 2D or 3D geometry. The PDEs and boundary values can be represented by two forms. The coefficient form is as follows:

$$d_a \frac{\partial u}{\partial t} - \nabla \cdot (c \nabla u + \alpha u - \gamma) + \beta \nabla u + a u = f$$

in Ω (1a)

$$\underline{n} \cdot (c \nabla u + \alpha u - \gamma) + q u = g - \lambda$$

on $\partial \Omega$ (1b)

$$h u = r$$

on $\partial \Omega$ (1c)

The first equation (1a) is satisfied inside the domain Ω and the second (1b) (*generalized Neumann boundary*) and third (1c) (*Dirichlet boundary*) equations are both satisfied on the boundary of the domain $\partial \Omega$. \underline{n} is the outward unit normal and is calculated internally. λ is an unknown vector-valued function called the *Lagrange multiplier*. This multiplier is also calculated internally and will only be used in the case of mixed boundary conditions. The coefficients d_a , c , α , β , γ , a , f , g , q and r are scalars, vectors, matrices or tensors. Their components can be functions of the space, time and the solution u . For a stationary system in coefficient form $d_a = 0$. Often c is called the *diffusion coefficient*, α and β are *convection coefficients*, a is the *absorption coefficient* and γ and f are *source terms*.

The second form of the PDEs and boundary conditions is the *general* form:

$$d_a \frac{\partial u}{\partial t} + \nabla \cdot \Gamma = F$$

in Ω (2a)

$$- \underline{n} \cdot \Gamma = G + \lambda$$

on $\partial \Omega$ (2b)

$$R = 0$$

on $\partial \Omega$ (2c)

The coefficients Γ and F can be functions of the space, time, the solution u and its gradient. The components of G and R can be functions of the space, time, and the solution u .

Given the geometry data, an initial finite element mesh is automatically generated by triangulation of the domain. The mesh is used for the discretisation of the PDE problem and can be modified to improve accuracy. The geometry, PDEs and boundary conditions are defined by a set of fields similarly to the structure in the language C. Also the boundary conditions can be functions of the space,

time, the solution u and its gradient. A graphical user interface can be used to simplify the input of these data. For solving purposes FemLab contains specific solvers (like static, dynamic, linear, non-linear solvers) for specific PDE problems. More detailed information on this subject can be found in (COMSOL, 2000).

3. A COMPLETE EXAMPLE

This example from (van Schijndel, 2003a; Sinha, 2000) deals with the velocity and temperature distribution in a room heated by a warm air stream. In figure 1 the geometry and boundary conditions are presented. The problem is modeled by 2D incompressible flow using the Boussinesq approximation with constant properties for the Reynolds and Grasshof numbers. The general form (2abc) is used for this type of non-linear problem. In figure 2, the PDE model, the corresponding PDE coefficients of (2a and 2c) and the corresponding part of the code are given. In (Sinha, 2000) the problem is solved and validated with measurements for several combinations of Re and Gr. In figure 3 the results for the temperature distributions are presented. The left-hand side shows the results obtained by (Sinha, 2000) and the right side show the corresponding FemLab results.

The test cases in this section show that FemLab is very reliable even for a highly non-linear problem such as convective airflow. Furthermore, for all the simulation results presented in this paper, the default mesh generation and solver are used. So good results can be obtained without a deep understanding of the gridding and solving techniques. The simulation time (based on processor Pentium 3, 500 MHz) are of order ~hours for highly non linear problems in the general form such as the airflow problem. Complete mfiles, results and movies can be found at:

<http://www.euro.femlab.com/support/knowledgebase/item.php?id=118&page=1&search=118>

Some limitations on FemLab CFD modeling

- 3D modeling is possible, however for practical usage, it consumes too much simulation time.
- The only present turbulence model at this moment is k-epsilon, other models are not developed yet.

Benefits of FemLab modeling

Scientists who want to create their own models can benefit from the combined PDEs modeling, the compact code, the easy adaptation of models, the state of the art solvers and the graphical output. Engineers can easily use these already developed models for other geometry's and boundary conditions. The FemLab models can be exported and connected with MatLab/Simulink models,

creating a flexible simulation environment for combined PDE and ODE (ordinary differential equation) based models (see the following Section).

4. AIRFLOW AND CONTROLLER

This section shows in detail how an airflow model of FemLab can be coupled to a controller of Simulink. FemLab has standard facilities to export models to SimuLink (van Schijndel, 2002; COMSOL, 2000). However, a standard export of the airflow model to SimuLink is not very practical because such a non-linear model can not be simulated efficiently by SimuLink (the standard solvers of SimuLink can not handle such a problem very well). A possible solution to this problem is to implement the FemLab code in the discrete section of a Simulink S-function. The S-function solves each time step (in this case 1 sec) an airflow problem using the FemLab solver. After each time step the solution is exported. Dependent on the controller, different boundary values can be applied. The use of the implementation of FemLab in a S-function is demonstrated in the following example. Again the FemLab model of Section 3 is used. The airflow at the inlet is now controlled by an on/off controller with hysteresis (Relay) of Simulink. If the temperature of the sensor is above 20.5 °C the air temperature at the inlet is 17 °C if the air temperature is below 19.5 °C the inlet temperature is 22 °C. Initially the inlet temperature is 18 °C. Figure 4 gives an overview of the results. Figure 5 shows details of how a S-function should be programmed for the use with this FemLab model and solver. More Details about S-functions can be found in (Mathworks, 1999). The above results show that also high non-linear models solved with FemLab can be exported to Simulink by writing an appropriate S-function. The complete mfiles, results and movies can be found at:

<http://www.euro.femlab.com/support/knowledgebase/item.php?id=119&page=1&search=119>

Benefits:

Engineers can use this model to study the effect of controller type and settings on the indoor airflow and temperature distribution.

5. OTHER DEVELOPMENTS

In this Section some other recent developments in the area of integrated building physics with FemLab/ SimuLink/MatLab are discussed. In order to integrate HAM models, a multi zone indoor climate model in SimuLink has been developed:

A multi zone indoor climate HAM model

In (van Schijndel, 2003c), a building energy and moisture performance model in the

Matlab/SimuLink environment is presented. The model consists of a continuous part with a variable time step and a discrete part with a time step of one hour. The main advantages of this numeric hybrid approach are: a) the dynamics of the building systems of time scales less than an hour are accurately simulated, b) the model becomes very time efficient as the discrete model uses 1 hour time steps. A yearly based simulation takes 5 minutes on a Pentium III, 500 MHz computer, c) the model can easily be integrated with SimuLink and FemLab models as discussed below.

The integration of the building systems is shown at:

Integration of Building systems and controllers

Also in (van Schijndel, 2003c), complete examples of the implementation of the following building systems in SimuLink are demonstrated: a) a heat pump, b) a thermal energy storage system and c) an energy roof system. Validation results are also included.

The following step is to integrate a detailed 2D (or 3D) construction of FemLab with the indoor climate model, using the standard facilities

Integration of an indoor climate model, a thermal bridge model and a controller

The objective of this subtask was to study the standard export facilities of FemLab to SimuLink. A 2D thermal bridge model has been exported using the standard export facilities of FemLab and placed into the room model. Not only detailed temperatures of the thermal bridge were simulated, but also the influence of the thermal bridge on the total heat flow of indoor climate. The model was successfully used as demonstration model for students to simulate the interaction between building construction (the thermal bridge), the indoor climate and controller settings. It showed that a linear model in FemLab could easily be exported to Simulink. The simulation time stays within limits.

Integration of an indoor climate model, a moisture transport model and a controller

(van Schijndel, 2003b) Presents the characteristics of the Walloon Church in Delft (Netherlands) and a description of constraints for the indoor climate, giving criteria for the indoor air temperature and relative humidity with the focus on the preservation of the monumental organ. The setpoint operation of the HVAC system is evaluated by simulation using MatLab, FemLab and Simulink models. The following main model components are presented and combined in a single integrated SimuLink model: 1) a WaVo Simulink building model (van

Schijndel, 2003c) for simulating the indoor temperature and relative humidity, 2) a FemLab PDE model for simulating detailed dynamic moisture transport in the monumental wood (organ) and 3) a Simulink controller model. Each model is separately validated with measurements. The main advantage of the integrated model is that it directly simulates the impact of HVAC control setpoint strategies on the indoor climate and the organ.

CONCLUSIONS

The combination FemLab/Simulink/Matlab is evaluated as solver for HVAC problems based on partial differential equations (PDEs). An example of the implementation of a typical HVAC modeling problem such as a dynamic airflow in combination with a controller is demonstrated. The simulation results of the presented airflow model show a good agreement with measurements. Other presented models show that the combination FemLab/Simulink/MatLab is a powerful and flexible environment for modeling and solving HVAC problems.

Benefits for scientists are:

- * The wide application area. The software is designed to simulate systems of coupled PDEs, 1D, 2D or 3D, non-linear and time dependent.
- * There is a clear relation between the mathematical model and (compact) program code in terms of specified PDE coefficients and boundary conditions.
- * The easy-to-use default gridders and solvers give accurate solutions.
- * The graphical output capabilities.

Benefits for engineers are:

- * Already developed Simulink models and controllers can be connected with FemLab models.
- * It is relatively easy to combine or adapt models for solving typical engineering problems.
- * It is relatively easy to create Graphical User Interfaces (GUI) of the Simulink models.

Benefits for designers are:

The FemLab/Simulink/MatLab environment facilitates state of the art PDE modeling with user-friendly Graphical User Interfaces.

REFERENCES

- COMSOL AB, 2000, FEMLAB Version 2.0 pre, Reference Manual, September 2000
- CSTB, SIMBAD, 2001, Building and HVAC Toolbox, Reference Manual, November 2001

Schijndel, 2003a, A.W.M. van, Modeling and solving building physics problems with FemLab, Building and Environment 38, pp319-327

Schijndel, A.W.M. van et al, 2003b, Optimal setpoint operation of the climate control of a monumental church, accepted for publication for the 2ND international conference on research in building physics, Leuven, September 14-18

Schijndel, A.W.M. van & Wit, M.H. de , 2003c Advanced simulation of building systems and control with SimuLink, 8TH IBPSA Conference Eindhoven. August 11-14.

Schijndel, A.W.M. van, 2002, Advanced HVAC modeling with FemLab/SimuLink/MatLab, Proc. 6TH Int. Conference System Simulation in Buildings Liege, December 16-18

Sinha, S.L., et al, 2000, Numerical simulation of two-dimensional room air flow with and without buoyancy, *Energy and Buildings* 32(2000) 121-129

The Mathworks, 1998 Inc. MatLab Reference Manual, Version 5.3.

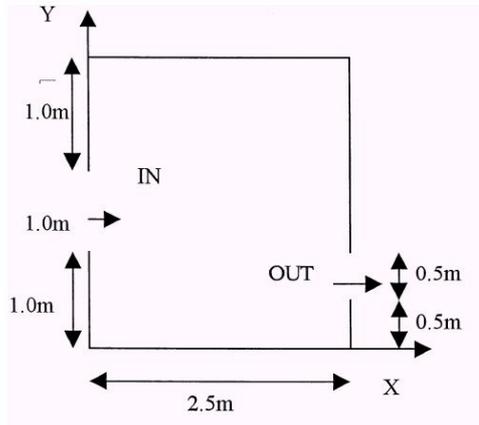
The Mathworks, 1999, Inc. Simulink Reference Manual, Version 3.

NOMENCLATURE

- A,B constants
 Gr Grasshof number [-]
 \underline{n} outward unit normal [-]
 p pressure [Pa]
 Pr Prandl number [-]
 Re Reynolds number [-]
 t time [s]
 T temperature [°C], ([-] when scaled)
 u solution of the PDE(s), e.g. temperature, moisture content, etc.
 u,v velocity (scaled) [-]
 x,y,z positions [m]
 λ Lagrange multiplier
 Ω domain
 $\partial\Omega$ boundary of domain

PDE Coefficients

- $a, c, d_a, f, F, g, G, h, q, r, R, \alpha, \beta, \gamma, \Gamma$



The boundary conditions are:

At the left, right, top and bottom walls:
 $u=0, v=0, T=0$.

At the inlet:
 $u=1, v=0, T=1$.

At the outlet :
 Neuman conditions for u, v and T

Figure 1. The geometry and boundary conditions for the 2D-airflow problem.

U-momentum equation

$$\frac{\partial u}{\partial t} = -\frac{\partial(uu)}{\partial x} - \frac{\partial(vu)}{\partial y} - \frac{\partial p}{\partial x} + \frac{1}{\text{Re}} \nabla^2 u$$

V-momentum equation

$$\frac{\partial v}{\partial t} = -\frac{\partial(uv)}{\partial x} - \frac{\partial(vv)}{\partial y} - \frac{\partial p}{\partial y} + \frac{1}{\text{Re}} \nabla^2 v + \frac{Gr}{\text{Re}^2} T$$

Continuity equation

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$$

Energy equation

$$\frac{\partial T}{\partial t} = -\frac{\partial(uT)}{\partial x} - \frac{\partial(vT)}{\partial y} + \frac{1}{\text{Re Pr}} \nabla^2 T$$

```

%Variables: u1=u; u2=v; u3=p u4=T; u1x = du1/dx etc.

fem.geom=poly2([0 0 0 0 2.5 2.5 2.5 2.5],[0 1 2 3 3 1 0.5 0]); %Specify yeometry
fem.mesh=meshinit(fem); fem.mesh=meshrefine(fem); %Mesh init and refine
fem.dim=4; fem.form='general'; %4 variables, general form
Re=1000;Gr=2.5e7; Pr=0.71; %Model parameters
etaval=1/Re; betaval=Gr/(Re*Re); alphaval=1/(Re*Pr);
fem.variables={'eta' etaval 'alpha' alphaval 'beta' betaval }; %Model variables

fem.bnd.r={{ '-u1'; '-u2'; 0; '0-u4' } ... %Specify boundary coefficient r
           { '-u1'; '-u2'; 0; '0-u4' } ...
           { '1-u1'; '-u2'; 0; '1-u4' } ...
           { '-u1'; '-u2'; 0; '0-u4' } ...
           { '-u1'; '-u2'; 0; '0-u4' } ...
           { '-u1'; '-u2'; 0; '0-u4' } ...
           { 0; 0; '-u3'; 0 } ...
           { '-u1'; '-u2'; 0; '0-u4' } };

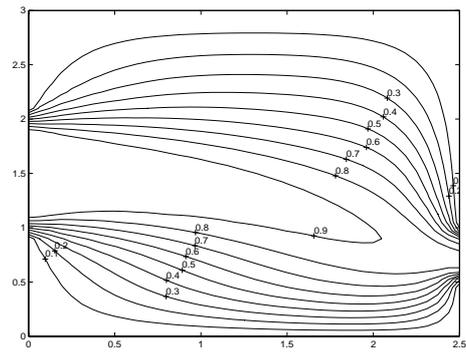
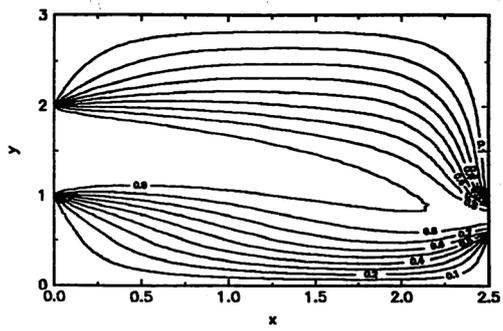
fem.equ.da={{ {1; 1; 0; 1} }; %Specify PDE coefficient da
fem.equ.f={{ '-(u1.*u1x+u2.*u1y+u3x)';... %Specify PDE coefficient f
             '-(u1.*u2x+u2.*u2y+u3y)+beta*u4';...
             '-(u1x+u2y)';...
             '-(u1.*u4x+u2.*u4y) '}}};

fem.equ.ga={{ { '-eta*u1x'; '-eta*u1y' };... %Specify PDE coefficient ga
              { '-eta*u2x'; '-eta*u2y' };...
              0;...
              { '-alpha*u4x'; '-alpha*u4y' }}};

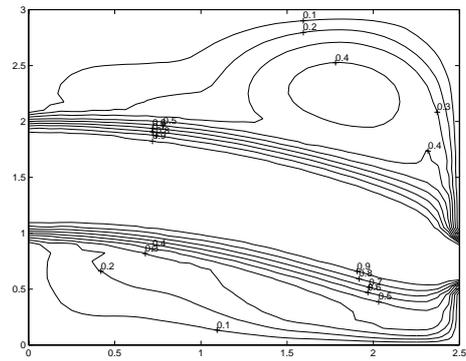
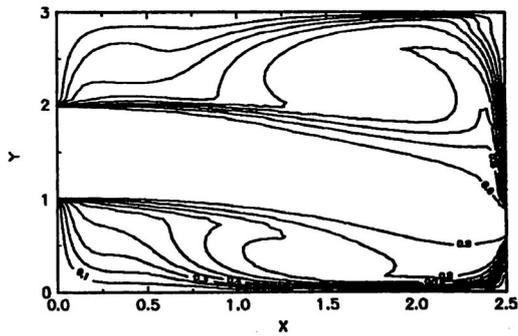
fem=femdiff(fem); %Calculate divergence gamma on domain
fem.init={{0 0 0 0}}; % Initial values
fem.sol=femtime(fem,'atol',{.1,.1,Inf,.1}),'ode','fldae',... %Solving
           'sd','on','report','on','tlist',0:.3:30);

```

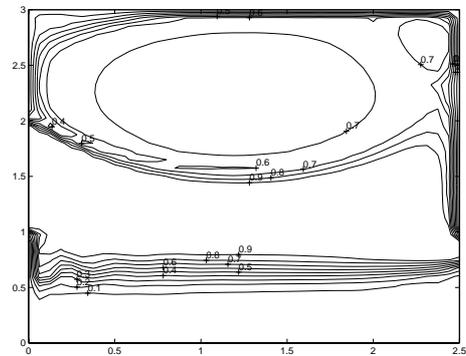
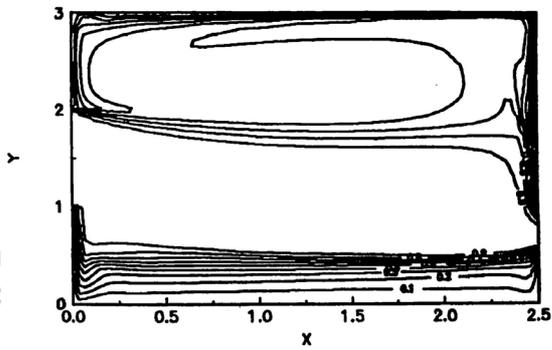
Figure 2. The PDE model and the corresponding FemLab code.



(a) $Re = 50, Gr = 0$



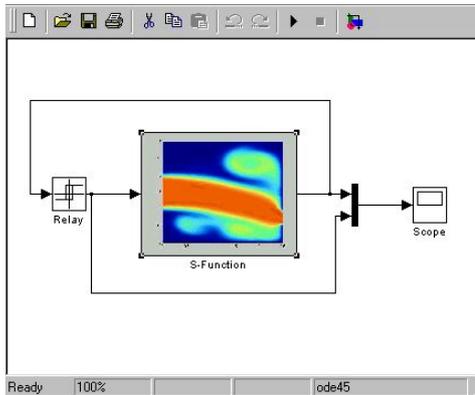
(b) $Re=1000, Gr=0$



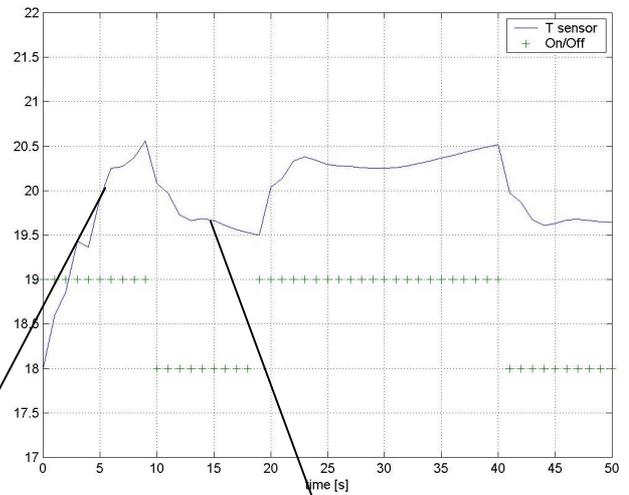
(c) $Re=1000, Gr=2.5 \cdot 10^7$

Figure 3. Dimensionless temperature contours comparison of the validated simulation results of (Sinha, 2000) (left hand side) with the FemLab results (right hand side) for the 2D-airflow problem.

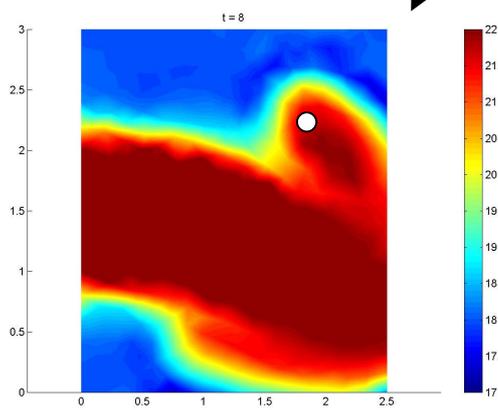
PART A



PART B



PART C



PART D

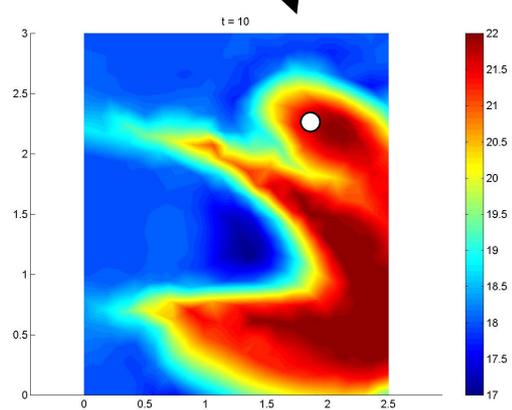


Figure 4. Part A, The Simulink model including the S-function of the FemLab model and controller (Relay), Part B, the temperature of the sensor (-) and the output of the on/off controller (+) versus time, Part C, the air temperature distribution at 8 sec. (hot air is blown in) and Part D, the air temperature distribution at 10 sec. (cold air is blown in) (o = sensor position).

```

function [sys,x0,str,ts] = dsfun(t,x,u,flag)
% t: time; x = state vector ; u = input vector;
% flag = control parameter by SimuLink
global OutT Outsol
% specify variable 'fem': the FemLab model geomtry, mesh,
% PDE coefficents and          boundary conditions

switch flag,
    case 0,
        [sys,x0,str,ts] = mdlInitializeSizes(fem,tstap,u0);% Initialization %
    case 2,
        sys = mdlUpdate(t,x,u,fem,tstap); %update discrete part
    case 3,
        sys = mdlOutputs(t,x,u,fem,tstap); %produce output
    case 9,
        sys = []; %terminate

end %end dsfunc

function [sys,x0,str,ts] = mdlInitializeSizes(fem,tstap,u0)
global OutT Outsol
% calculated the next values needed for SimuLink
% calculated initial values for time OutT(1) and
% initial values for solution Outsol(1)
% x0 are all values of the temperature on the mesh
OutT(1)=0;
Outsol(1)=x0;

function sys = mdlUpdate(t,x,u,fem,tstap)
global OutT Outsol
if u(1)>0.5 % if dimensionless Temperature of sensor > 0.5 then
    fem.bnd.r={{ '-u1'; '-u2' ;0 ; '0-u4' } ... %boundary value cold air
else
    fem.bnd.r={{ '-u1'; '-u2' ;0 ; '1-u4' } ... % bounadry value hot air
end
tijd=[t; t+tstap/2; t+tstap]; %calculate simulatetime for FemLab model
fem.sol=femtime(fem,'atol',{.1,.1,Inf,.1}),'ode','fldae',...
    'sd','on','report','on','tlist',tijd,'init',x); %simulate airflow
OutT=[OutT (t+tstap)]; % update time
Outsol=[Outsol fem.sol.u(:,ntijd)]; % update solution

function sys = mdlOutputs(t,x,u,fem,tstap)
global OutT Outsol
Points=[69 84 101 167 188 211 259 261]; % location of sensor in mesh
sys=mean(x(Points)); % output

```

Figure 5. Example of the implementation of a FemLab model in the discrete section of a S-function of SimuLink