Temperature Distribution in a Vacuum Flask

Introduction

The following example solves for the temperature distribution within a vacuum flask holding hot coffee. The main interest here is to illustrate how to use MATLAB® functions to define material properties and boundary condition directly within the COMSOL model.

Two MATLAB functions are used to define the temperature dependent thermal conductivity of the vacuum flask shell and the insulation foam, while a third function defines the heat transfer coefficient that corresponds to a natural convective cooling for a vertical plate and surrounding air.

Model Definition

Assume axial symmetry for this simulation to reduce the model geometry to the 2D cross-section of the vacuum flask geometry shown in Figure 1.

The vacuum flask consists of a steel shell isolated with a foam material, and a cork made of nylon. On the inside wall apply a constant temperature, assuming that the vacuum flask is filled with coffee of constant temperature.

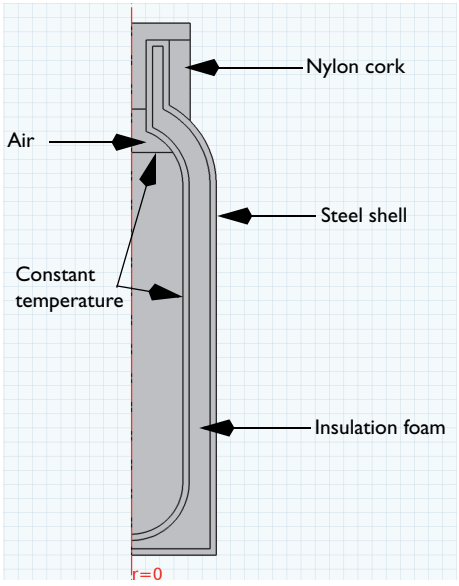

Figure 1: Cross section of the vacuum flask geometry

Define the temperature dependent thermal conductivity of steel according to the following polynomial expression:

$$k_{\text{steel}} = 71.12 - 0.115T + 1.16e^{-4}T^2 - 4.25e^{-8}T^3$$

As shown in Figure 2, in the temperature interval of interest for the model, the thermal conductivity decreases with increasing temperature.

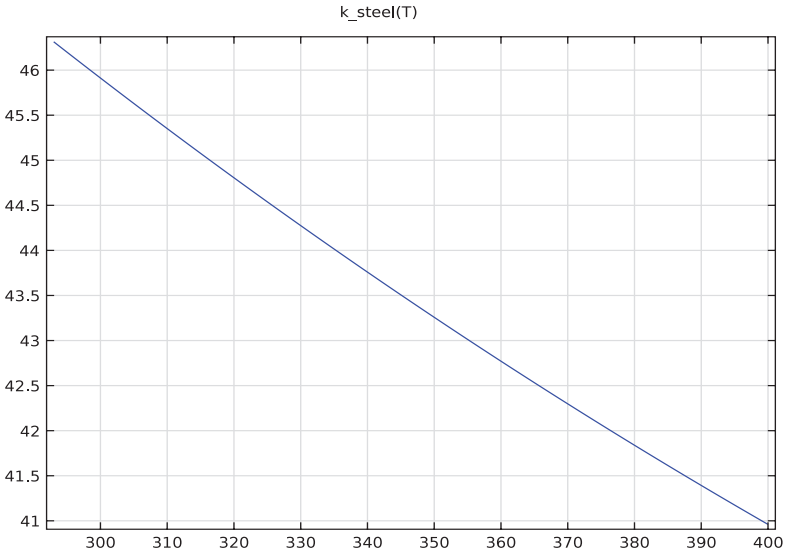

Figure 2: Thermal conductivity of the steel wall versus temperature.

Use a different polynomial expression for the thermal conductivity of the insulating foam according to:

$$k_{\text{foam}} = -2.141 + 1.87e^{-2}T - 5.3e^{-5}T^2 + 4.945e^{-8}T^3$$

The above expression is plotted in Figure 3 for the temperature interval 293 K – 400 K.

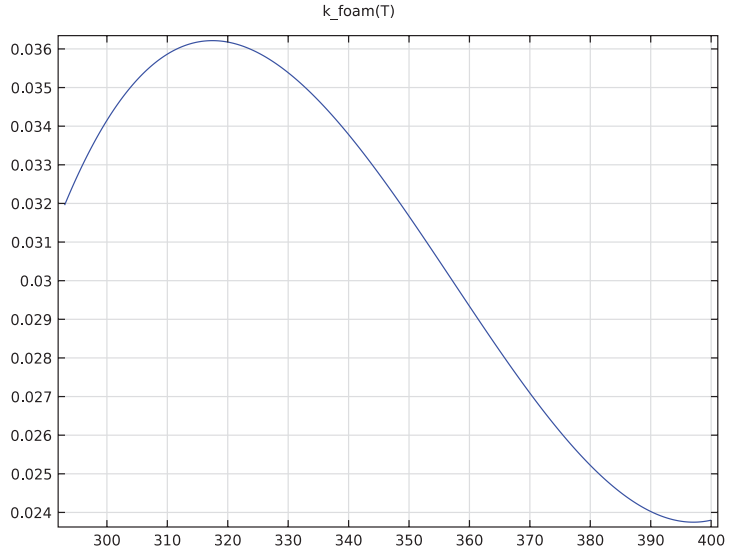

Figure 3: Thermal conductivity of the insulating foam versus temperature.

The outer surface dissipates heat via natural convection. This loss is characterized by the convective heat transfer coefficient, h, which in practice you often determine with empirical handbook correlations. Because these correlations depend on the surface temperature, T_{surface}, engineers must estimate T_{surface} and then iterate between h and T_{surface} to obtain a converged value for h. The following expressions is a typical handbook correlation for h for the case of natural convection in air on a vertical heated wall:

$$h = \frac{k \cdot \text{Nu}}{L}$$

$$\text{Nu} = \left(825e^{-3} + \frac{387e^{-3} \cdot \text{Ra}^{\frac{1}{6}}}{\left(1 + \left(\frac{492e^{-3}}{\text{Pr}}\right)^{\frac{9}{16}}\right)^{\frac{8}{27}}} \right)^2$$

$$\text{Ra} = 2g\left(\frac{\rho^2 Cp}{\eta k}\right)\left|\frac{T_{\text{surface}} - T_{\text{ambient}}}{T_{\text{surface}} + T_{\text{ambient}}}\right| L^3$$

In the above equation T_{ambient} is the temperature of the air surrounding the vacuum flask, and all other variables are defined in the function script, under step 10 in Modeling Instructions—MATLAB®.

The figure below illustrates the expression of heat transfer coefficient as function of the wall temperature, the ambient temperature and the wall length:

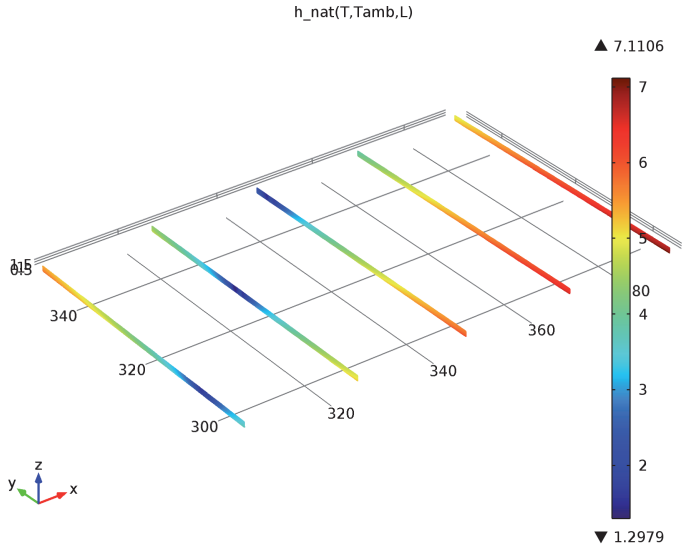

Figure 4: Heat transfer coefficient for natural convection cooling of a vertical wall. The x-axis represents the surface temperature, T_{surface}, the y-axis represents the ambient temperature, T_{ambient}, and the z-axis represents the wall length, L.

Results and Discussion

From the temperature distribution in the vacuum flask wall shown in Figure 5 you can see that most of the temperature gradients are in the foam, which shows that the material works well for insulating the vacuum flask.

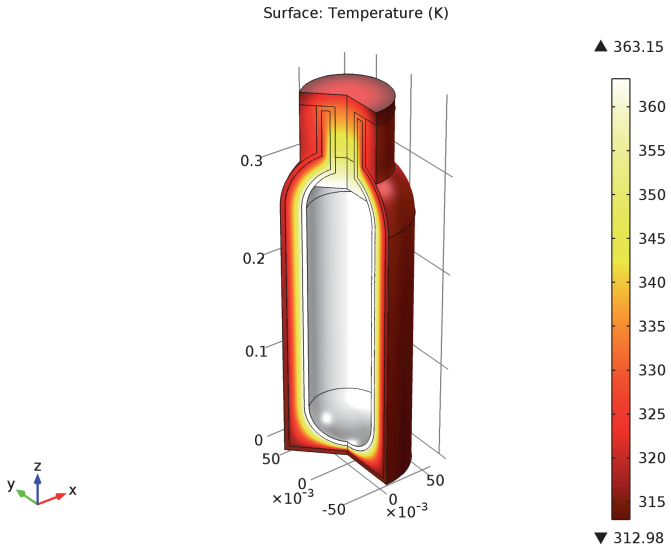

Figure 5: Temperature distribution in the vacuum flask.

Figure 6 shows the heat transfer coefficient along the vacuum flask surface. Its value depends on the temperature surface, and the discontinuity in the curve is at the boundary between the cap and the steel wall.

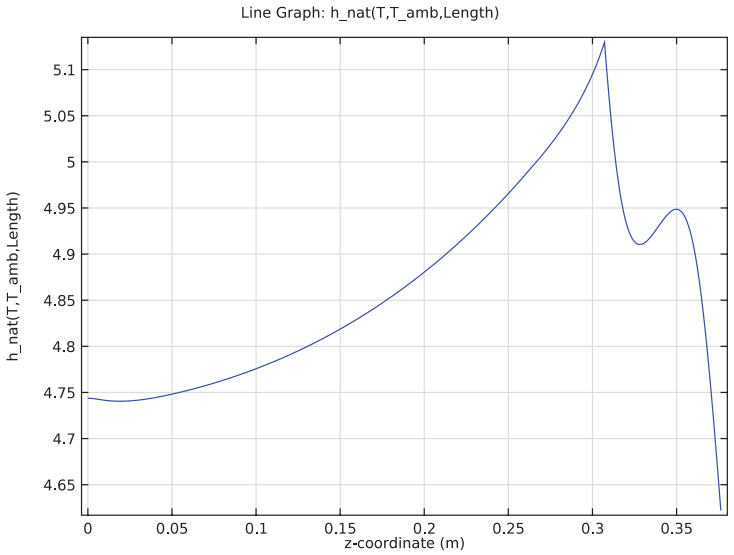

Figure 6: Heat transfer coefficient along the surface of the vacuum flask.

Notes About the COMSOL Implementation

To use an external MATLAB function in a COMSOL model add a MATLAB feature node to the tree in the model builder, and specify the functions there. Once you solve the model, COMSOL automatically starts a MATLAB process that evaluates the functions and returns the value to the COMSOL model.

Model Library path: `COMSOL_Multiphysics/LiveLink_for_MATLAB/Tutorial_Models/vacuum_flask_llmatlab`

Modeling Instructions—MATLAB®

Before the implementation of the model in the COMSOL Desktop®, define the MATLAB functions for the thermal conductivity of steel and insulating foam, and the convective heat transfer coefficient.

1 Start the MATLAB editor or any text editor.

2 In a new file enter the following:

```
function out = k_steel(T)
% k_steel returns a value for the steel thermal conductivity
% that vary with the local temperature
out = 71.12-0.115*T+1.16e-4*T.^2-4.25e-8*T.^3;
```

3 Save the file as `k_steel.m`.

4 Define now the function derivative with respect to T. Open a new file and enter the lines below:

```
function out = k_steel_dT(T)
% Derivative with respect of T of the function k_steel
out = -0.115+2.32e-4*T-12.75e-8*T.^2;
```

5 Save the file as `k_steel_dT.m`.

6 Specify now the function for the thermal conductivity of the foam material. Open a new file and enter the following commands:

```
function out = k_foam(T)
% k_foam returns a value for the insulation foam thermal
% conductivity that vary with the local temperature
out = -2.141+1.87e-2*T-5.3e-5*T.^2+4.945e-8*T.^3;
```

7 Save the file as `k_foam.m`.

8 Define the function derivative with respect to T. Open a new file and enter the following lines:

```
function out = k_foam_dT(T)
% Derivative with respect of T of the function k_foam
out = 1.87e-2-10.6e-5*T+14.835e-8*T.^2;
```

9 Save the file as `k_foam_dT.m`.

10 Finally, create one more file with the lines below to define the heat transfer coefficient for the cooling by natural convection:

```
function out = h_nat(T, Tamb, L)
% h_nat returns the value of the heat transfer coefficient
% corresponding to the cooling of a vertical plate by air in
% natural convection. The heat transfer coefficient depends
% on the surface temperature T, the temperature of the
% surrounding air Tamb and the length of the of the wall

% Air properties
k = 2e-2; % Thermal condunctivity [W/(m*K)]
rho = 1; % Density [kg/m^3]
cp = 1e3; % Heat capacity [J/(kg*K)]
mu = 1e-6; % Dynamic viscosity [Pa*s]
g = 9.81; % Gravity [m/s^2]
```

```
% Definition of Prandt number
Pr = cp*mu/k;
% Definition of Rayleigh number
Ra = 2*g*rho^2*cp/(mu*k).*abs((T-Tamb)./(T+Tamb)).*L.^3;
% Defifinition of Nusselt number
num = 387e-3*Ra.^(1/6);
denom = (1+(492e-3/Pr)^(9/16))^(8/27);
Nu = (825e-3+num/denom).^2;
% Return the heat transfer coefficient value
out = k.*Nu./L;
```

II Save the file as `h_nat.m`.

Note: It is important to have the same size for the function output as for the input argument. As COMSOL Multiphysics assembles the problem in blocks, the variables have the size of an array. To prevent possible size related issues, use point wise operators (`.*`, `./`, `.^`) with the input arguments.

SETTING THE DIRECTORY PATH IN MATLAB

To be able to solve the model in the COMSOL Desktop, you need to make sure that the path to the MATLAB functions that are used in the model is set in MATLAB. Refer to the section Setting the Function Directory Path in MATLAB in the *LiveLink™ for MATLAB® User's Guide* for details on how do to this.

Modeling Instructions—COMSOL Desktop

From **File** menu, choose **New**.

NEW

I In the **New** window, click the **Model Wizard** button.

MODEL WIZARD

1 In the **Model Wizard** window, click the **2D Axisymmetric** button.

2 In the **Select physics** tree, select **Heat Transfer>Heat Transfer in Solids (ht)**.

3 Click the **Add** button.

4 Click the **Study** button.

5 In the tree, select **Preset Studies>Stationary**.

6 Click the **Done** button.

GLOBAL DEFINITIONS

Parameters

1 On the **Home** toolbar, click **Parameters**.

2 In the **Parameters** settings window, locate the **Parameters** section.

3 In the table, enter the following settings:

Name	Expression	Value	Description
Length	38[cm]	0.38000 m	Height of the vacuum flask
T_amb	20[degC]	293.15 K	Temperature of surrounding air

MATLAB 1

1 On the **Home** toolbar, click **Functions** and choose **Global>MATLAB**.

2 In the **MATLAB** settings window, locate the **Functions** section.

3 In the table, enter the following settings:

Function name	Arguments
k_steel	T
k_foam	T

4 Click to expand the **Plot parameters** section. In the table, enter the following settings:

Lower limit	Upper limit
293	400

5 Click the **Plot** button to get the same figure as Figure 2.

By default only the first function in the list that has all argument defined in the plot parameters section is displayed. If you want to display the second function you can use the **Move up** button.

6 In the **Function** column click k_foam then under the table click **Move Up**.

7 Click the **Plot** button to get the same figure as Figure 3.

8 In the table, enter the following settings:

Function name	Arguments
h_nat	T, Tamb, L

To display the function `h_nat` as in Figure 4, add the plot parameters definition for `Tamb` and `L`.

9 In the table, add the following plot limit settings:

Lower limit	Upper limit
293	350
0.5	1.5

10 Click the **Plot** button.

Now define the derivative of the functions `k_steel` and `k_foam` with respect to the temperature `T`.

11 Click to expand the **Derivatives** section. In the table, enter the following settings:

Function name	Argument	Partial derivative
k_foam	T	k_foam_dT(T)
k_steel	T	k_steel_dT(T)

12 In the table, enter the following settings:

Function name	Arguments
k_steel_dT	T
k_foam_dT	T

GEOMETRY 1

Import 1

1 On the **Geometry** toolbar, click **Import**.

2 In the **Import** settings window, locate the **Import** section.

3 Click the **Browse** button.

4 Browse to the model's Model Library folder and double-click the file `vacuum_flask_llmatlab.dxf`.

5 Click the **Import** button.

Delete Entities 1

1 In the **Model Builder** window, right-click **Geometry 1** and choose **Delete Entities**.

2 In the **Delete Entities** settings window, locate the **Entities or Objects to Delete** section.

3 From the **Geometric entity level** list, choose **Domain**.

4 On the object **imp1**, select Domain 3 only.

Form Union

In the **Model Builder** window, under **Component 1>Geometry 1** right-click **Form Union** and choose **Build Selected**.

MATERIAL BROWSER

1 On the **Home** toolbar, click **Browse Materials**.

2 In the **Material Browser** window, In the tree, select **Built-In>Nylon**.

3 Click the **Add to Component** button.

4 In the tree, select **Built-In>Air**.

5 Click the **Add to Component** button.

6 Click the **Close** button.

MATERIALS

Nylon

1 In the **Model Builder** window, under **Component 1>Materials** click **Nylon**.

2 In the **Material** settings window, locate the **Geometric Entity Selection** section.

3 In the **Selection** list, choose **1**, **2**, and **3**.

4 Click **Remove from Selection**.

Air

1 In the **Model Builder** window, under **Component 1>Materials** click **Air**.

2 Select Domain 3 only.

Material 3

1 On the **Home** toolbar, click **New Material**.

2 Select Domain 1 only.

3 In the **Material** settings window, locate the **Material Contents** section.

4 In the table, enter the following settings:

Property	Name	Value	Unit	Property group
Thermal conductivity	k	`k_steel(T)`	W/(m·K)	Basic
Density	rho	`7850`	kg/m³	Basic
Heat capacity at constant pressure	Cp	`475`	J/(kg·K)	Basic

5 Right-click **Component 1>Materials>Material 3** and choose **Rename**.

6 Go to the **Rename Material** dialog box and type `Steel` in the **New name** edit field.

7 Click **OK**.

Material 4

1 On the **Home** toolbar, click **New Material**.

2 Select Domain 2 only.

3 In the **Material** settings window, locate the **Material Contents** section.

4 In the table, enter the following settings:

Property	Name	Value	Unit	Property group
Thermal conductivity	k	`k_foam(T)`	W/(m·K)	Basic
Density	rho	`24`	kg/m³	Basic
Heat capacity at constant pressure	Cp	`2.3`	J/(kg·K)	Basic

5 Right-click **Component 1>Materials>Material 4** and choose **Rename**.

6 Go to the **Rename Material** dialog box and type `Foam` in the **New name** edit field.

7 Click **OK**.

HEAT TRANSFER IN SOLIDS

Temperature 1

1 On the **Physics** toolbar, click **Boundaries** and choose **Temperature**.

2 Select Boundaries 7, 20, 27, and 32 only.

3 In the **Temperature** settings window, locate the **Temperature** section.

4 In the T_0 edit field, type `90[degC]`.

Heat Flux 1

1 On the **Physics** toolbar, click **Boundaries** and choose **Heat Flux**.

2 Select Boundaries 11, 22, 23, 25, and 33 only.

3 In the **Heat Flux** settings window, locate the **Heat Flux** section.

4 Click the **Inward heat flux** button.

5 In the h edit field, type `h_nat(T,T_amb,Length)`.

6 In the T_{ext} edit field, type `T_amb`.

MESH 1

1 In the **Model Builder** window, under **Component 1** click **Mesh 1**.

2 In the **Mesh** settings window, locate the **Mesh Settings** section.

3 From the **Element size** list, choose **Extremely fine**.

4 Click the **Build All** button.

SAVE THE MODEL

Before computing the solution you can save the model in the same folder as for the M-function. This will ensure you that the function path is automatically known by the MATLAB engine.

1 From the upper left menu ( ) (Windows users) or from the Options menu (Mac and Linux users), select **Save**.

2 Browse to the folder where you have previously saved the M-functions.

3 In the File name field, enter `vacuum_flask_llmatlab` in the **File name** edit field.

4 Click **Save**.

STUDY 1

On the **Home** toolbar, click **Compute**.

RESULTS

Figure 5 is shown as first default plot.

1D Plot Group 3

1 On the **Home** toolbar, click **Add Plot Group** and choose **1D Plot Group**.

2 On the **1D plot group 3** toolbar, click **Line Graph**.

3 Select Boundaries 22, 23, 25, and 33 only.

4 In the **Line Graph** settings window, locate the **y-Axis Data** section.

5 In the **Expression** edit field, type `h_nat(T,T_amb,Length)`.

6 Locate the **x-Axis Data** section. From the **Parameter** list, choose **Expression**.

7 In the **Expression** edit field, type `z`.

8 On the **1D plot group 3** toolbar, click **Plot**.