TNSolver Version 0.9.x
User Manual

Robert J. Cochran
Applied Computational Heat Transfer [1]
Seattle, Washington
TNSolver@heattransfer.org

Document Version 0.9.x
Working Draft Processed using LaTeX 2_ε on July 30, 2016

Contents

List of Tables

List of Figures

Nomenclature

Command Description Character Symbols

- {} list of valid parameters
- <> default parameter in the list of parameters
- | separator for the list of valid parameters
- (I) single integer number
- (I . . .) list of integer numbers
- (R) single real number
- (R . . .) list of real numbers
- (S) single character string
- (S . . .) list of character strings

Chapter 1

Introduction

TNSolver is a thermal network solver. This document describes how to set up and run a thermal analysis model using TNSolver. GNU Octave [EBHW15] was used to develop the solver, using the MATLAB programming language. It will run in either Octave or MATLAB and is an open source program.

1.1 Heat Transfer Math Model

Thermal network models are based on the integral form of the math model for heat transfer. The concept of a control volume is defined by a volume with a surface. The control volume surface is assumed to be stationary in the following, hence a fixed control volume.

The energy conservation equation, in Cartesian tensor integral form, in terms of temperature, T, is:

$$\underbrace{\frac{d}{dt}\iint_V \rho c_v T \, dV}_{\text{capacitance}} + \underbrace{\int_A q_i n_i \, dA}_{\text{conduction}} + \underbrace{\int_A \rho c_p u_i T n_i \, dA}_{\text{advection}} = \underbrace{\iint_V \dot{q} \, dV}_{\text{source}} \tag{1.1}$$

where u_i is the fluid flow advection velocity and the specific heat $c_p = c_v = c$ for a solid. Note that the advection velocity field, u_i, must conform to the integral conservation of mass equation:

$$\frac{d}{dt}\iint_V \rho \, dV + \int_A \rho u_i n_i \, dA = 0 \tag{1.2}$$

1.1.1 Conduction

The steady conduction equation with no advection, in Cartesian tensor integral form, is:

$$\int_A q_i n_i \, dA = \iint_V \dot{q} \, dV \tag{1.3}$$

where Fourier's Law of Heat Conduction for an anisotropic thermal conductivity, k_{ij}, provides a constitutive model for the heat flux as a function of temperature gradient:

$$q_i = -k_{ij} \frac{\partial T}{\partial x_j} \tag{1.4}$$

For the case of an isotropic thermal conductivity, k, the heat flux model simplifies to:

$$q_i = -k \frac{\partial T}{\partial x_i} \tag{1.5}$$

1.1.2 Convection

Convection heat transfer at the surface of the control volume is given by:

$$\int_{\Gamma_c} q_i n_i \, dA = \int_{\Gamma_c} h(T_s - T_c) \, dA, \quad \text{where} \begin{cases} T_s > T_c, & \text{cooling} \\ T_s < T_c, & \text{heating} \end{cases} \tag{1.6}$$

The convection coefficient, $h(x_i, t, T_s, T_c)$, is usually a function of position, time, surface temperature, free stream or bulk temperature and other parameters. The value of the coefficient is evaluated using a correlation.

1.1.3 Radiation

The surface radiation boundary condition models the case when a surface is radiating to a far field surface ($F_{s-r} = 1.0$):

$$\int_{\Gamma_r} q_i n_i \, dA = \int_{\Gamma_r} \sigma \epsilon_s (T_s^4 - T_r^4) \, dA, \quad \text{where} \begin{cases} T_s > T_r, & \text{cooling} \\ T_s < T_r, & \text{heating} \end{cases} \tag{1.7}$$

where ϵ_s is the emissivity of the surface and σ is the Stefan-Boltzmann constant ($\sigma = 5.67040 \times 10^{-8}$ $W/m^2 \cdot K^4$, SI units or $\sigma = 1.714 \times 10^{-9}$ $Btu/hr \cdot ft^2 \cdot {}^\circ R^4$, English units). Note that the temperature must be the absolute temperature.

1.1.4 Advection

1.2 Overview of the Input File Structure

1.3 Document Status

During the development stage of TNSolver, this document will be in a constant state of change, as well as being incomplete. As the first full version of TNSolver is approached, this user manual will be complete, with the help of feedback from early TNSolver users.

Document Version 0.1.0/TNSolver Version 0.4.0 This is the first release of the user manual for TNSolver. It contains descriptions of all the commands used to define a thermal model. At this version of TNSolver the support for US units is sketchy, at best. It is recommended to stick with SI units for this release.

Chapter 2

Solution Parameters

The `Solution Parameters` command block is used to set parameters associated with the thermal model.

```
Begin Solution Parameters

  title                         = (S ...)
  type                          = {<steady>|transient}
  units                         = {<SI>|US}
  T units                       = {<C>|K|F|R}
  nonlinear convergence         = (R)
  maximum nonlinear iterations = (I)
  begin time                    = (R)
  end time                      = (R)
  number of time steps          = (I)
  Stefan-Boltzmann              = 5.6704e-8 (W/m^2)
  gravity                       = 9.80665 (m/s^2)
  graphviz output               = {<no>|yes}

End Solution Parameters
```

2.1 `title`

```
  title = (S ...)
```

Type Line Command
Scope `Solution Parameters` block
Parameter (S . . .) **Default** None

Description

This line command defines a title for the thermal network model.

Example

A thermal network model of the heat transfer through a plane wall is given the title ”Heat transfer through a plane wall”:

```
Begin Solution Parameters
  ...
  title = Heat transfer through a plane wall
  ...
End Solution Parameters
```

2.2 type

```
type = {<steady>|transient}
```

Type	Line Command		
Scope	`Solution Parameters` block		
Parameter	`{<steady>|transient}`	**Default**	`steady`

Description

This line command will set the type of thermal model to steady or transient. A steady thermal model will not have any thermal capacitance terms. A transient thermal model requires defining a start and end time for the simulation.

Example

A transient simulation is desired:

```
Begin Solution Parameters
  ...
  type = transient
  ...
End Solution Parameters
```

2.3 units

```
units = {<SI>|US}
```

Type Line Command
Scope `Solution Parameters` block
Parameter {<SI>|US} **Default** `SI`

Description

This line command will set the units for the simulation.

Example

SI units are desired:

```
Begin Solution Parameters
  ...
  units = SI
  ...
End Solution Parameters
```

2.4 T units

```
T units = {<C>|K|F|R}
```

Type Line Command
Scope `Solution Parameters` block
Parameter {<C>|K|F|R} **Default** `C`

Description

This line command will set whether absolute temperatures will be used in the input and output files or not. All temperatures in the output will be given the specified units.

Example

Absolute temperatures are desired for both the input file, as well as all the output, for a simulation using SI units:

```
Begin Solution Parameters
  ...
  T units = K
  ...
End Solution Parameters
```

2.5 nonlinear convergence

```
nonlinear convergence = {<1.0E-9>|(R)}
```

Type Line Command
Scope Solution Parameters block
Parameter {<1.0E-9>|(R)} **Default** 1.0E-9

Description

The residual of the energy conservation equation is used to determine the convergence of the nonlinear thermal model. The residual is nondimensionalized using the L2 norm of the absolute temperature:

$$\frac{R}{\sqrt{\sum_{k=1}^{n} T_k^2}} \tag{2.1}$$

Example

The nonlinear convergence is set to 10^{-10}:

```
Begin Solution Parameters
  ...
  nonlinear convergence = 1.0e-10
  ...
End Solution Parameters
```

2.6 maximum nonlinear iterations

```
maximum nonlinear iterations = {<100>|(I)}
```

Type Line Command
Scope Solution Parameters block
Parameter {<100>|(I)} **Default** 100

Description

This line command will set the maximum number of nonlinear iterations for a simulation. For the case of a steady simulation, the simulation will stop when the number of nonlinear iterations reaches this number. For a transient simulation, a time step will end when this number is reached.

Example

The maximum number of nonlinear iterations is set to 15:

```
Begin Solution Parameters
  ...
  maximum nonlinear iterations = 15
  ...
End Solution Parameters
```

2.7 begin time

```
begin time = {<0.0>|(R)}
```

Type Line Command
Scope `Solution Parameters` block
Parameter `{<0.0>|(R)}` **Default** `0.0`

Description

This line command will set the initial time for a transient simulation.

Example

The initial time for a simulation is 12.3:

```
Begin Solution Parameters
  ...
  begin time = 12.3
  ...
End Solution Parameters
```

2.8 end time

```
end time = (R)
```

Type Line Command
Scope `Solution Parameters` block
Parameter (R) **Default** None

Description

This line command will set the ending time for a transient simulation.

Example

The end time for a simulation is 350.0:

```
Begin Solution Parameters
  ...
  end time = 350.0
  ...
End Solution Parameters
```

2.9 time step

```
time step = (R)
```

Type	Line Command		
Scope	`Solution Parameters` block		
Parameter	(R)	**Default**	None

Description

This line command will set the time step, Δt, for a transient simulation. Either the time step or number of times steps is required for a transient simulation. If the time step is specified, the number of time steps is given by:

$$n = \frac{t_{\text{end}} - t_{\text{begin}}}{\Delta t} \tag{2.2}$$

Example

A time step of 0.1 is desired for a transient simulation:

```
Begin Solution Parameters
  ...
  time step = 0.1
  ...
End Solution Parameters
```

2.10 number of time steps

```
number of time steps = (I)
```

Type Line Command
Scope `Solution Parameters` block
Parameter (I) **Default** None

Description

This line command will specify the number of time steps to take for a transient simulation. The time step size, Δt, is given by:

$$\Delta t = \frac{t_{\text{end}} - t_{\text{begin}}}{n} \tag{2.3}$$

Example

The number of time steps for a transient simulation is set to 50:

```
Begin Solution Parameters
  ...
  number of time steps = 50
  ...
End Solution Parameters
```

2.11 `print interval`

```
print interval = {<1>|(I)}
```

Type Line Command
Scope `Solution Parameters` block
Parameter `{<1>|(I)}` **Default** `1`

Description

This line command will set the output interval for a transient simulation.

Example

An output interval of 5 is set for a transient simulation:

```
Begin Solution Parameters
  ...
  print interval = 5
  ...
End Solution Parameters
```

2.12 Stefan-Boltzmann

```
Stefan-Boltzmann = {<5.6704E-8 W/m^2-K^4>|1.714e-9 Btu/hr-ft^2-R^4}
```

Type Line Command
Scope `Solution Parameters` block
Parameter `{<5.6704E-8>|1.714E-9}` **Default** `5.6704E-8`

Description

This line command will set the Stefan-Boltzmann constant, σ, for use in radiation heat transfer. The default value is $5.6704 \times 10^{-8}\,\mathrm{W/m^2 - K^4}$ for SI units and $1.714 \times 10^{-9}\,\mathrm{Btu/hr - ft^2 - R^4}$ for US units.

Example

Set the Stefan-Boltzmann constant for SI units:

```
Begin Solution Parameters
  ...
  Stefan-Boltzmann = 5.6704E-8
  ...
End Solution Parameters
```

2.13 gravity

```
gravity = {<9.80665 m/s^2>|32.174 ft/s^2}
```

Type Line Command
Scope `Solution Parameters` block
Parameter `{<9.80665>|32.174}` **Default** `9.80665`

Description

This line command sets the value of gravity for the simulation. When the units are set to SI, the default value for gravity is $9.80665 m/s^2$. When the units are set to US, the default value for gravity is $32.174 ft/s^2$.

Example

Set the gravity to 9.806:

```
Begin Solution Parameters
  ...
  gravity = 9.806
  ...
End Solution Parameters
```

2.14 graphviz output

```
graphviz output = {<no>|yes}
```

Type	Line Command		
Scope	`Solution Parameters` block		
Parameter	`{<no>|yes}`	**Default**	`no`

Description

This line command will request that a Graphviz dot file be generated for the solution. See Graphviz - Graph Visualization Software, for the open source program that will visualize the thermal network solution in the dot file.

Example

Turn on the output of a graphviz dot file:

```
Begin Solution Parameters
  ...
  graphviz output = yes
  ...
End Solution Parameters
```

Chapter 3

Nodes

The nodes command block is used to set the volume and material properties for temperature nodes in the model. Surface nodes, by definition, have zero volume. The material property can be specified or given by the name of a material from the material library.

```
Begin Nodes

! label  material  volume
   (S)     (S)      (R)

! label  density*specific heat  volume
   (S)          (R)              (R)

End Nodes
```

Chapter 4

Conductors

The `Conductors` command block is used to define the conductors in the thermal network model. There are four broad classes of conductors. They are conduction, convection, radiation and advection. For temperature dependent material properties, an entry in the material library is required (see Chapter 10).

```
Begin Conductors

! label    type        nd_i  nd_j  parameters
   (S)  conduction    (S)   (S)   (R) (R) (R)      ! k, L, A
   (S)  conduction    (S)   (S)   (S) (R) (R)      ! material, L, A
   (S)  cylindrical   (S)   (S)   (R) (R) (R) (R)  ! k, ri, ro, L
   (S)  cylindrical   (S)   (S)   (S) (R) (R) (R)  ! material, ri, ro, L
   (S)  spherical     (S)   (S)   (R) (R) (R)      ! k, ri, ro
   (S)  spherical     (S)   (S)   (S) (R) (R)      ! material, ri, ro

   (S)  convection    (S)   (S)   (R) (R)          ! h, A
   (S)  IFCduct       (S)   (S)   (S) (R) (R) (R)  ! material, velocity, Dh, A
   (S)  EFCcyl        (S)   (S)   (S) (R) (R) (R)  ! material, velocity, D, A
   (S)  EFCdiamond    (S)   (S)   (S) (R) (R) (R)  ! material, velocity, D, A
   (S)  EFCimpjet     (S)   (S)   (S) (R) (R) (R) (R) ! material, velocity, D,
                                                      ! H, r
   (S)  EFCplate      (S)   (S)   (S) (R) (R) (R) (R) ! material, velocity,
                                                      ! Xbegin, Xend, A
   (S)  EFCsphere     (S)   (S)   (S) (R) (R)      ! material, velocity, D
   (S)  ENChcyl       (S)   (S)   (S) (R) (R)      ! material, D, A

   (S)  ENChplatedown (S)   (S)   (S) (R) (R)      ! material, L=A/P, A
   (S)  ENChplateup   (S)   (S)   (S) (R) (R)      ! material, L=A/P, A
   (S)  ENCiplatedown (S)   (S)   (S) (R) (R) (R) (R)  ! material, H, L=A/P,
                                                       ! angle, A
   (S)  ENCiplateup   (S)   (S)   (S) (R) (R) (R) (R)  ! material, H, L=A/P,
                                                       ! angle, A
   (S)  ENCsphere     (S)   (S)   (S) (R)          ! material, D
```

```
   (S)  ENCvplate     (S)   (S)  (S) (R) (R)      ! material, L, A
   (S)  FCuser        (S)   (S)  (S) (S) (R...) (R)  ! function, material,
                                                     ! parameters, A
   (S)  NCuser        (S)   (S)  (S) (S) (R...) (R)  ! function, material,
                                                     ! parameters, A

   (S)  surfrad       (S)   (S)  (R) (R)          ! emissivity, A
   (S)  radiation     (S)   (S)  (R) (R)          ! script-F, A

   (S)  advection     (S)   (S)  (S) (R) (R)      ! material, velocity, A
   (S)  outflow       (S)   (S)  (S) (R) (R)      ! material, velocity, A

End Conductors
```

4.1 Conduction

These conductors are used to model the diffusive transport of energy. There are three different geometries supported: 1) Cartesian or planar wall, 2) cylindrical and 3) spherical.

4.1.1 `conduction`

```
! label    type        nd_i  nd_j  parameters
   (S)  conduction    (S)   (S)  (R) (R) (R)      ! k, L, A
   (S)  conduction    (S)   (S)  (S) (R) (R)      ! material, L, A
```

Type Line Command
Scope `Conductors` block
Parameters $<k$ |material $>$, L, A **Default** None

Description

Cartesian conduction heat transfer:

$$Q_{ij} = \frac{kA}{L}(T_i - T_j) \tag{4.1}$$

where k is the thermal conductivity, L is the length and A is the area, see Figure 4.1. The thermal conductivity can be given or a material library entry specified.

Example

Conduction through a steel planar wall which is 0.23 m thick and has an area of 2 m^2:

```
Begin Conductors
```

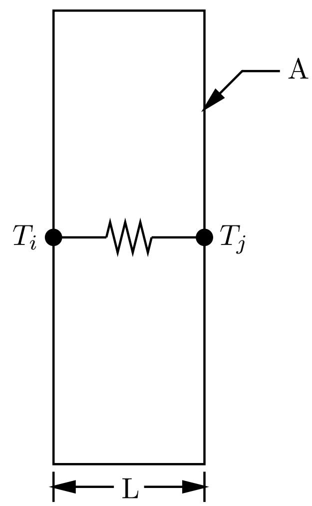


Figure 4.1: Cartesian Conduction

```
   ...
! label    type       nd_i  nd_j  parameters
  wall  conduction    in    out   steel, 0.23,  2    ! material, L, A
   ...
End Conductors
```

4.1.2 `cylindrical`

```
! label    type       nd_i  nd_j  parameters
   (S)  cylindrical   (S)   (S)  (R) (R) (R) (R)  ! k, ri, ro, L
   (S)  cylindrical   (S)   (S)  (S) (R) (R) (R)  ! material, ri, ro, L
```

Type Line Command
Scope `Conductors` block
Parameters <k |material >, r_i, r_o, L **Default** None

Description

The radial heat transfer rate between the inner radius, r_i, temperature, T_i and the outer radius, r_o, temperature, T_o, is:

$$Q_{io} = \frac{k2\pi L}{\ln(r_o/r_i)}(T_i - T_o) \tag{4.2}$$

where k is the thermal conductivity and L is the length of the cylinder, see Figure 4.2. The thermal conductivity can be given or a material library entry specified.

Example

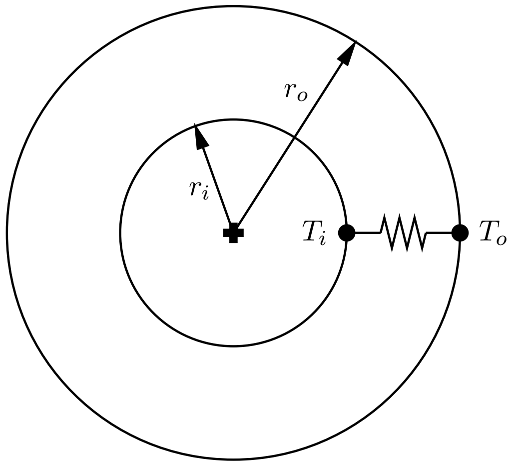


Figure 4.2: Cylindrical Conduction

Conduction through a 3 m long copper pipe wall with an inner radius of 0.05 m and an outer radius of 0.055 m is:

```
Begin Conductors
  ...
! label    type      nd_i  nd_j  parameters
  pipe  cylindrical   in    out  383.0 0.05  0.055  3  ! k, ri, ro, L
  ...
End Conductors
```

4.1.3 spherical

```
! label    type       nd_i  nd_j  parameters
   (S)  spherical     (S)   (S)  (R) (R) (R)     ! k, ri, ro
   (S)  spherical     (S)   (S)  (S) (R) (R)     ! material, ri, ro
```

Type Line Command
Scope `Conductors` block
Parameters <k |material >, r_i, r_o **Default** None

Description

The radial heat transfer rate between the inner radius, r_i, temperature, T_i and the outer radius, r_o, temperature, T_o, is:

$$Q_{io} = \frac{k 4\pi r_o r_i}{r_o - r_i}(T_i - T_o) \tag{4.3}$$

where k is the thermal conductivity, see Figure 4.3. The thermal conductivity can be given or a material library entry specified.

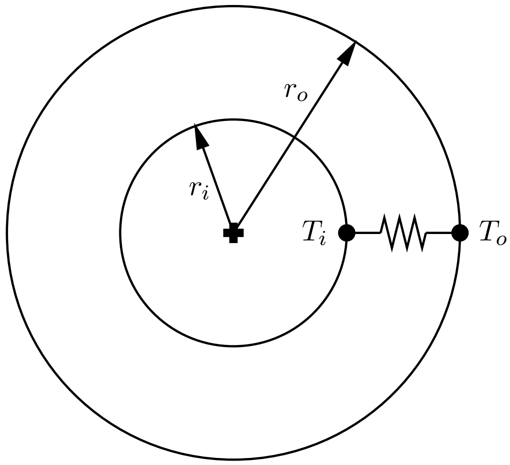


Figure 4.3: Spherical Conduction

Example

The conduction heat transfer through a hollow sphere is required. The inner radius of the sphere is $r_i = 0.03\ m$ and the outer radius is $r_o = 0.045$. The thermal conductivity is $k = 43.7 W/m \cdot K$. The conductor is labeled `sphere` and the inner surface is `in` and the outer surface node is `out`:

```
Begin Conductors
  ...
! label    type        nd_i  nd_j  parameters
  sphere  spherical   in    out    43.7 0.03 0.04   ! k, ri, ro
  ...
End Conductors
```

4.2 Convection

Convection heat transfer is primarily modeled using correlations. A variety of correlations are provided for both forced and natural convection scenarios, see Table 4.1.

Note: Due to the nature of fluid property evaluations for the various correlations, the first node in the conductor needs to be the surface node, T_s and the second node needs to be the fluid node, T_∞, in all cases.

4.2.1 `convection`

```
! label    type        nd_i  nd_j  parameters
   (S)   convection    (S)   (S)   (R) (R)          ! h, A
```

Table 4.1: Available Convection Correlations

Geometry	Conductor	Flow Range	
Internal Forced Convection (`IFC`)			
pipe/duct	`IFCduct`	$0 \lesssim Re \lesssim 5\times10^6$	$0.5 \lesssim Pr \lesssim 2000$
External Forced Convection (`EFC`)			
cylinder	`EFCcyl`	$0.4 \lesssim Re \lesssim 400,000$	$0.7 \lesssim Pr$
diamond/square	`EFCdiamond`	$6,000 \lesssim Re \lesssim 60,000$	$0.7 \lesssim Pr$
impinging round jet	`EFCimpjet`	$6,000 \lesssim Re \lesssim 60,000$	$0.7 \lesssim Pr$
flat plate	`EFCplate`	$0 \lesssim Re \lesssim 10^8$	$0.6 \lesssim Pr \lesssim 60$
sphere	`EFCsphere`	$3.5 \lesssim Re \lesssim 7.6\times10^4$	$0.71 \lesssim Pr \lesssim 380$
Internal Natural Convection (`INC`)			
rectangular enclosure	`INCvenc`	$10^3 \lesssim Ra \lesssim 10^{10}$	$0.7 \lesssim Pr \lesssim 10^5$
External Natural Convection (`ENC`)			
horizontal cylinder	`ENChcyl`	$Ra \lesssim 10^{12}$	$0.7 \lesssim Pr$
horizontal plate facing down	`ENChplatedown`	$10^4 \lesssim Ra \lesssim 10^{11}$	$0.7 \lesssim Pr$
horizontal plate facing up	`ENChplateup`	$10^4 \lesssim Ra \lesssim 10^{11}$	$0.7 \lesssim Pr$
inclined plate facing down	`ENCiplatedown`	$Ra \lesssim 10^{11}$	$0.7 \lesssim Pr$
inclined plate facing up	`ENCiplateup`	$Ra \lesssim 10^{11}$	$0.7 \lesssim Pr$
sphere	`ENCsphere`	$Ra \lesssim 10^{11}$	$0.7 \lesssim Pr$
vertical flat plate	`ENCvplate`	$Ra \lesssim 10^{12}$	$0.7 \lesssim Pr$

Type Line Command
Scope `Conductors` block
Parameters h, A **Default** None
h is the convection coefficient
A is the surface area for convection

Description

Convection heat transfer, with a specified heat transfer coefficient, h:

$$Q_{ij} = hA(T_i - T_j) \tag{4.4}$$

where h is the specified convective heat transfer coefficient and A is the surface area of the control volume.

Example

The convective heat transfer from a wall is modeled using a conductor named `fluid`, with a surface node named `out` and connected to the free stream air temperature `Tc`. The convection coefficient is 5.4 W/m^2K and the surface area is 12 m^2:

```
Begin Conductors
  ...
! label    type       nd_i  nd_j  parameters
```

```
  fluid  convection   out   Tc    5.4  12      ! h, A
  ...
End Conductors
```

4.2.2 `IFCduct` Internal Forced Convection in a Duct/Pipe

```
! label    type     nd_i  nd_j  parameters
   (S)   IFCduct    (S)   (S)  (S) (R) (R) (R)  ! material, velocity, Dh, A
```

Type Line Command
Scope `Conductors` block
Parameters material, u, D_h, A **Default** None
u is the average fluid velocity
$D_h = 4 \times$ (Cross-sectional Area) /Wetted Perimeter $= 4A_c/P$ is the hydraulic diameter
$A =$ Duct Length $\times$ Perimeter $= LP$ is the surface area for convection

Description

For fully developed flow in a pipe or duct, the convection heat transfer is:

$$Q_{ij} = hA(T_s - T_\infty) \tag{4.5}$$

The heat transfer coefficient, h, is evaluated using the correlation and approach developed by Gnielinski (see [Gni13]): For the laminar flow regime, $Re \leq 2300$, the Nusselt number is:

$$\overline{Nu}_D = 3.66 \tag{4.6}$$

For fully turbulent flow, $Re \geq 4000$, the Nusselt number is:

$$\overline{Nu}_D = \frac{(f/8)(Re_D - 1000)Pr}{1 + 12.7\sqrt{f/8}(Pr^{2/3} - 1)} \tag{4.7}$$

where the friction factor, f, is given by:

$$f = (1.8 \log_{10} Re - 1.5)^{-2} \tag{4.8}$$

In the transition region, $2300 < Re < 4000$, a linear interpolation of the laminar and turbulent Nusselt numbers is used:

$$\overline{Nu}_D = (1 - \gamma)\overline{Nu}_{lam,2300} + \gamma\overline{Nu}_{turb,4000} \tag{4.9}$$

$$\gamma = \frac{Re - 2300}{4000 - 2300} \tag{4.10}$$

The Reynolds number is:

$$Re_D = \frac{\rho V D_h}{\mu} = \frac{V D_h}{\nu} \tag{4.11}$$

The fluid properties are evaluated at the film temperature:

$$T_f = \frac{T_s + T_\infty}{2} \tag{4.12}$$

Once the Nusselt number is determined, the convection coefficient, h, is given by:

$$h = \frac{\overline{Nu}_D k}{D_h} \tag{4.13}$$

where k is the thermal conductivity of the fluid. For the round pipe shown in Figure 4.4, the surface area for convection is:

$$A = L \times \pi \left(\frac{D}{2}\right)^2 \tag{4.14}$$

For the rectangular duct shown in Figure 4.4, the hydraulic diameter is:

$$D_h = \frac{4WH}{2(W+H)} = \frac{WH}{W+H} \tag{4.15}$$

and the surface area is:

$$A = L \times 2(W+H) \tag{4.16}$$

The heat transfer coefficient for air flowing in a 0.05 m diameter pipe is shown in Figure 4.5 for a

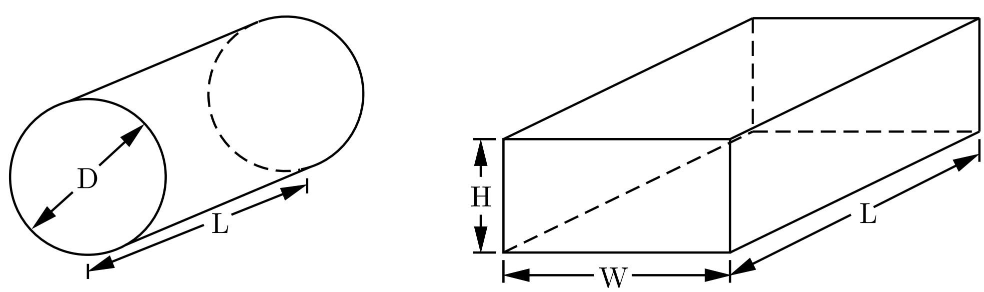


Figure 4.4: Pipe and Rectangular Duct Geometry

range of velocities.

Example

Forced convection heat transfer in a rectangular duct that is 0.2 m wide, 0.1 m high and 3 m long, with air flowing at an average velocity of 5 m/s, is modeled with a conductor labeled `duct`, with the wall node named `wall` and the fluid node named `fluid`:

```
Begin Conductors
   ...
! label    type   nd_i  nd_j  parameters
   duct  IFCduct  wall fluid  air 5.0 0.0666667 1.8  ! material, velocity, DH, A
   ...
End Conductors
```

4.2.3 `EFCcyl` External Forced Convection Over a Cylinder

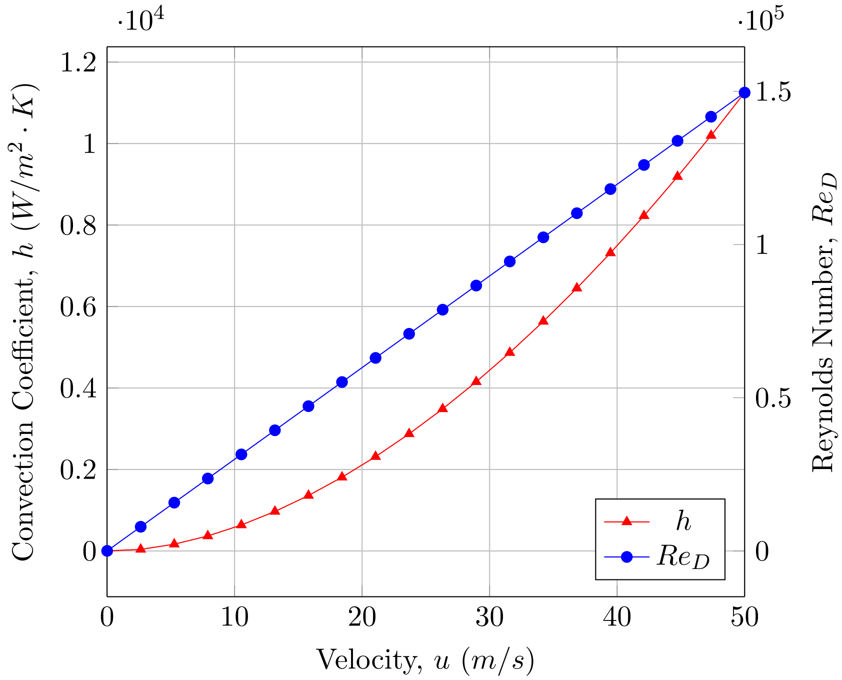


Figure 4.5: IFCduct, air, $D_h = 0.05\ m$, $T_f = 37.5\ C$

```
! label   type   nd_i  nd_j  parameters
  (S)    EFCcyl (S)   (S)   (S) (R) (R) (R)  ! material, velocity, D, A
```

Type Line Command
Scope `Conductors` block
Parameters material, u, D, A **Default** None
u is the fluid velocity
D is the diameter of the cylinder
$A = \pi D \times L$ is the surface area for convection

Description

External forced convection over a cylinder:

$$Q_{ij} = hA(T_s - T_\infty) \tag{4.17}$$

The heat transfer coefficient, h, is evaluated using the correlation (see Equation (7.44) , p. 436, in [BLID11] or [KK58]):

$$\overline{Nu}_D \equiv \frac{\bar{h}D}{k} = CRe_D^m Pr^{1/3} \tag{4.18}$$

where D is the diameter of the cylinder (see Figure 4.6) and the Reynolds number is:

$$Re_D = \frac{\rho V D}{\mu} = \frac{VD}{\nu} \tag{4.19}$$

The coefficients are given in Table 4.2 for a range of Reynolds number. If the Reynolds number is less than 0.4 or greater than 400,000 a warning is printed to the screen and the coefficient for 0.4 or 400,000 are used. The fluid properties are evaluated at the film temperature:

$$T_f = \frac{T_i + T_j}{2} \tag{4.20}$$

Once the Nusselt number is determined, the convection coefficient, h, is given by:

$$h = \frac{\overline{Nu_D} k}{D} \tag{4.21}$$

where k is the thermal conductivity of the fluid.

Table 4.2: External Forced Convection over a Cylinder

Re_D	C	m
0.4–4	0.989	0.330
4–40	0.911	0.385
40–4,000	0.683	0.466
4,000–40,000	0.193	0.618
40,000–400,000	0.027	0.805

Table 7.2, page 437 in [BLID11]

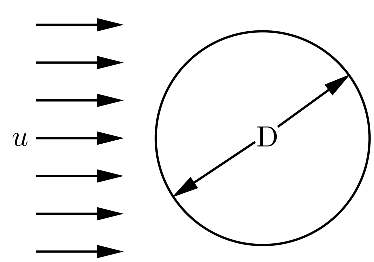


Figure 4.6: Geometry for Forced Convection over a Cylinder

Example

A conductor is required for forced convection heat transfer from a cylinder in cross-flow. The cylinder diameter is 0.08 m, and it is 0.9 m long, area is $A = \pi 0.08 \times 0.9 = 0.2262$. The fluid is water, with a velocity of 2 m/s. The conductor is given the label `fluid`, with a surface node called `wall` and the free stream temperature is called `Tinf`:

```
Begin Conductors
  ...
! label   type  nd_i  nd_j  parameters
  fluid  EFCcyl wall  Tinf  water  2.0  0.08 0.2262  ! material, velocity, D, A
  ...
End Conductors
```

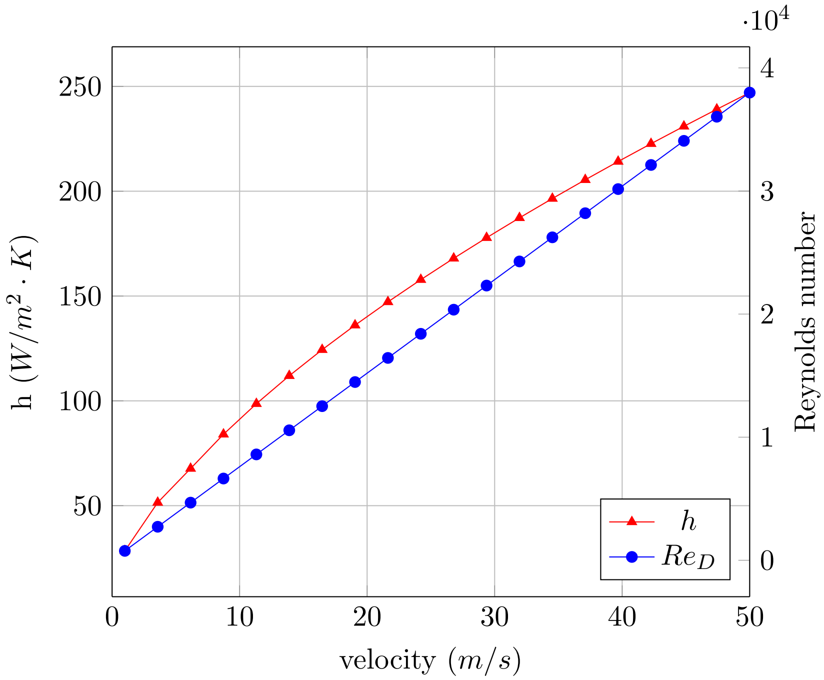


Figure 4.7: EFCcyl, air

4.2.4 `EFCdiamond` External Forced Convection Over a Diamond/Square

```
! label   type  nd_i  nd_j  parameters
   (S)   EFCdiamond (S)   (S)  (S) (R) (R) (R)  ! material, velocity, D, A
```

Type	Line Command		
Scope	`Conductors` block		
Parameters	material, u, D, A	**Default**	None
	u is the fluid velocity		
	$D = \sqrt{2W^2}$ is the characteristic length		
	$A = 4W \times L$ is the surface area for convection		

Description

External forced convection over a diamond/square:

$$Q_{ij} = hA(T_i - T_j) \tag{4.22}$$

The heat transfer coefficient, h, is evaluated using the correlation (see Equation (7.44), p. 436, in [BLID11]):

$$\overline{Nu}_D \equiv \frac{\bar{h}D}{k} = CRe_D^m Pr^{1/3} \tag{4.23}$$

where $D = \sqrt{2H^2}$ is the characteristic length (see Figure 4.8) and the Reynolds number is:

$$Re_D = \frac{\rho VD}{\mu} = \frac{VD}{\nu} \tag{4.24}$$

For the case of a gas flowing over noncircular cylinders in crossflow (Table 7.3, page 437 in [BLID11]): Note that the fluid properties are evaluated at the film temperature, T_f:

Table 4.3: Correlation Constants

Geometry	Re_D	C	m
⇉ ◆ $D = \sqrt{2W^2}$	6,000–60,000	0.304	0.59

$$T_f = \frac{T_s + T_\infty}{2} \tag{4.25}$$

Once the Nusselt number is determined, the convection coefficient, h, is given by:

$$h = \frac{\overline{Nu_D} k}{D} \tag{4.26}$$

where k is the thermal conductivity of the fluid. The heat transfer coefficient for a 1/2 inch diamond,

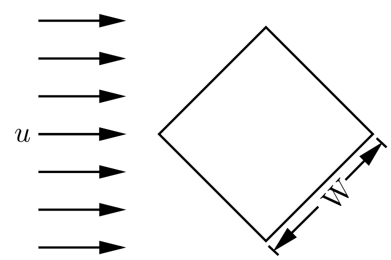


Figure 4.8: Geometry for Diamond in Cross-flow

in air, is shown in Figure 4.9 for a range of flow velocities.

Example

A conductor is required for forced convection heat transfer from a diamond in cross-flow. The width is $W = 0.0254$ m, $D = \sqrt{2 \times 0.0254^2} = 0.03592$ m, and it is 0.5 m long. The convection area is $A = 4.0254 \times 0.5 = 0.0508 m^2$. The fluid is air, with a velocity of 10 m/s. The conductor is given the label `fluid`, with a surface node called `surf` and the free stream temperature is called `Tinf`:

```
Begin Conductors
  ...
! label    type     nd_i nd_j parameters
  fluid  EFCdiamond surf Tinf air 10.0 0.03592 0.0508 ! material, velocity, D, A
  ...
End Conductors
```

4.2.5 `EFCimpjet` External Forced Convection of an Impinging Round Jet

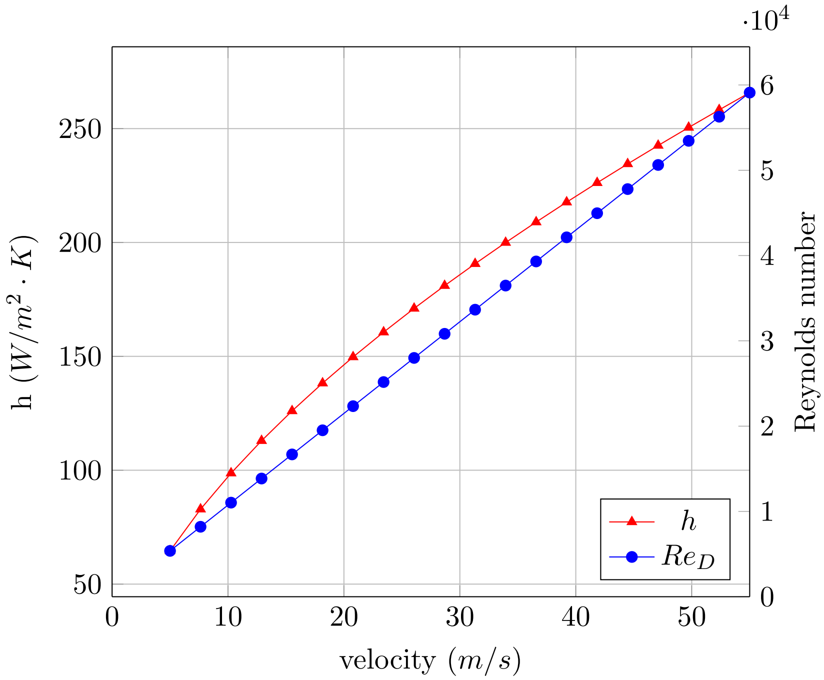


Figure 4.9: EFCdiamond, air, $T_f = 37.5\ C$, $W = 0.5$", $D = \sqrt{2W^2} = 0.017961\ m$

```
! label   type  nd_i  nd_j  parameters
   (S)   EFCimpjet (S)   (S)  (S) (R) (R) (R) (R) ! material, velocity, D, H, r
```

Type Line Command
Scope `Conductors` block
Parameters material, u, D, H, r **Default** None
u is the fluid velocity
D is the jet diameter
H is the height above plate
r is the radius for convection

Description

External forced convection of a single impinging round jet:

$$Q_{ij} = hA(T_i - T_j) \tag{4.27}$$

The heat transfer coefficient, h, is evaluated using the correlation (see Equation (7.62), p. 459, in [BLID11] or [Mar77]):

$$\overline{Nu}_D \equiv \frac{\bar{h}D}{k} = G\left(A_r, \frac{H}{D}\right)[2Re^{1/2}(1+0.005Re^{0.55})^{1/2}]Pr^{0.42} \tag{4.28}$$

where $A_r = D^2/4r^2$ and

$$G = 2A_r^{1/2}\frac{1-2.2A_r^{1/2}}{1+0.2(H/D-6)A_r^{1/2}} \tag{4.29}$$

where D is the diameter of the jet (see Figure 4.10) and the Reynolds number is:

$$Re_D = \frac{\rho V D}{\mu} = \frac{VD}{\nu} \tag{4.30}$$

Note that the fluid properties are evaluated at the film temperature, T_f:

$$T_f = \frac{T_s + T_\infty}{2} \tag{4.31}$$

Once the Nusselt number is determined, the convection coefficient, h, is given by:

$$h = \frac{\overline{Nu}_D k}{D} \tag{4.32}$$

where k is the thermal conductivity of the fluid. The heat transfer coefficient for a 1/2 inch diameter

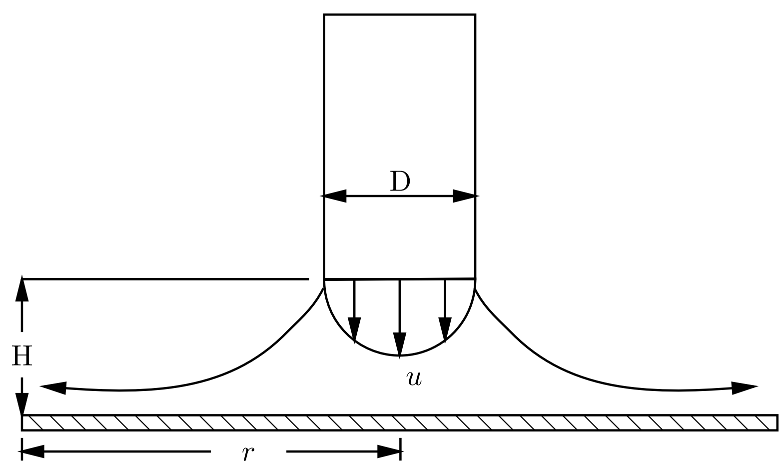


Figure 4.10: Geometry for Impinging Round Jet

air jet, is shown in Figure 4.11 for a range of flow velocities.

Example

A conductor is required for forced convection heat transfer from an impinging round jet. The radius is $r =\ m$. The fluid is air, with a velocity of 10 m/s. The conductor is given the label `jet`, with a surface node called `Tsurf` and the jet temperature is called `Tjet`:

```
Begin Conductors
  ...
! label  type     nd_i nd_j parameters
  jet  EFCimpjet Tsurf Tjet air 10.0 0.0127 0.127 0.0318 ! material, velocity, D, H, r
  ...
End Conductors
```

4.2.6 `EFCplate` External Forced Convection Over a Flat Plate

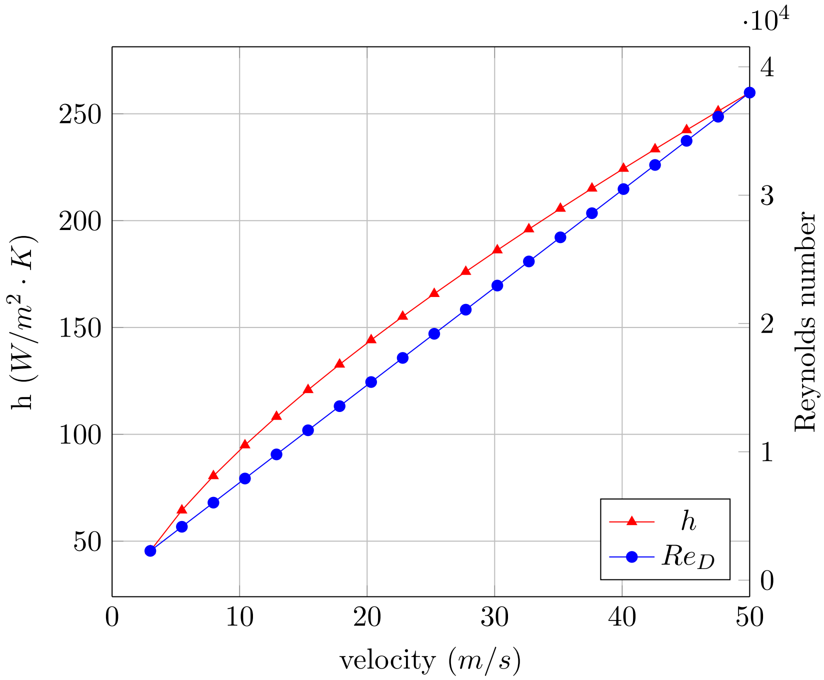


Figure 4.11: EFCimpjet, air, $D = 0.0127\ m$, $H = 0.127\ m$ and $r = 0.0318\ m$

```
! label   type   nd_i  nd_j  parameters
   (S) EFCplate (S)   (S)  (S) (R) (R) (R)  ! material, velocity, Xbeg, Xend, A
```

Type Line Command
Scope `Conductors` block
Parameters material, u, x_{begin}, x_{end}, A **Default** None
u is the free stream velocity
x_{begin} is the distance from the leading edge
x_{end} is the distance to the trailing edge
A is the area for convection

Description

External forced convection over a flat plate:

$$Q_{ij} = hA(T_s - T_\infty) \tag{4.33}$$

The average convection heat transfer coefficient is given by:

$$h = \frac{1}{L}\int_0^L h(x)dx \tag{4.34}$$

In general, the transition from laminar to turbulent flow occurs at $Re_{cr} = \frac{\rho u x_{cr}}{\mu} = 5 \times 10^5$ or

$x_{cr} = 5 \times 10^5 \left(\frac{\mu}{\rho u}\right)$. The local Nusselt number, Nu_x, for laminar flow is:

$$Nu_x = \frac{h_x x}{k} = 0.332 Re_x^{1/2} Pr^{1/3} \tag{4.35}$$

and the local Nusselt number, Nu_x, for turbulent flow is:

$$Nu_x = \frac{h_x x}{k} = 0.0296 Re_x^{4/5} Pr^{1/3} \tag{4.36}$$

Then:

$$h = \frac{1}{L}\left(\int_0^{x_{cr}} h_{x,\mathrm{laminar}} dx + \int_{x_{cr}}^{L} h_{x,\mathrm{turbulent}} dx\right) \tag{4.37}$$

or,

$$h = \frac{1}{L}\left[\ \int_0^{x_{cr}} \left(\frac{k}{x}\right) 0.332 Re_x^{1/2} Pr^{1/3} dx + \int_{x_{cr}}^{L} \left(\frac{k}{x}\right) 0.0296 Re_x^{4/5} Pr^{1/3} dx\right] \tag{4.38}$$

then if $x_{begin} > x_{cr}$:,

$$h = \left(\frac{k}{L}\right)\left[\ 0.332\left(\frac{\rho u}{\mu}\right)^{1/2} \int_0^{x_{cr}} \frac{1}{x^{1/2}} dx + 0.0296\left(\frac{\rho u}{\mu}\right)^{4/5} \int_{x_{cr}}^{L} \frac{1}{x^{1/5}} dx\right] Pr^{1/3} \tag{4.39}$$

$$h = \left(\frac{k}{L}\right)\left[\ 0.332\left(\frac{\rho u}{\mu}\right)^{1/2} 2x_{cr}^{1/2} + 0.0296\left(\frac{\rho u}{\mu}\right)^{4/5} \frac{5}{4}\left(L^{4/5} - x_{cr}^{4/5}\right)\right] Pr^{1/3} \tag{4.40}$$

$$h = \left(\frac{k}{L}\right)\left[\ 0.664\,\mathrm{Re}_{cr}^{1/2} + 0.037\left(\mathrm{Re}_L^{4/5} - \mathrm{Re}_{cr}^{4/5}\right)\right] Pr^{1/3} \tag{4.41}$$

for a transition Reynolds number of $Re_{cr} = 5 \times 10^5$, the heat transfer coefficient is:

$$h = \left(\frac{k}{L}\right)\left[0.037\,\mathrm{Re}_L^{4/5} - 871.3\right] Pr^{1/3} \tag{4.42}$$

Note that the fluid properties are evaluated at the film temperature, T_f:

$$T_f = \frac{T_s + T_\infty}{2} \tag{4.43}$$

The heat transfer coefficient for a $1m$ long flat plate, in air, is shown in Figure 4.13 for a range of

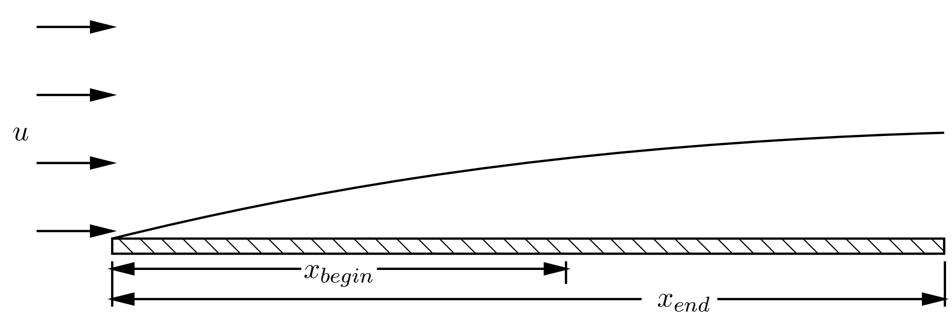


Figure 4.12: Geometry for Forced Convection from a Flat Plate

flow velocities.

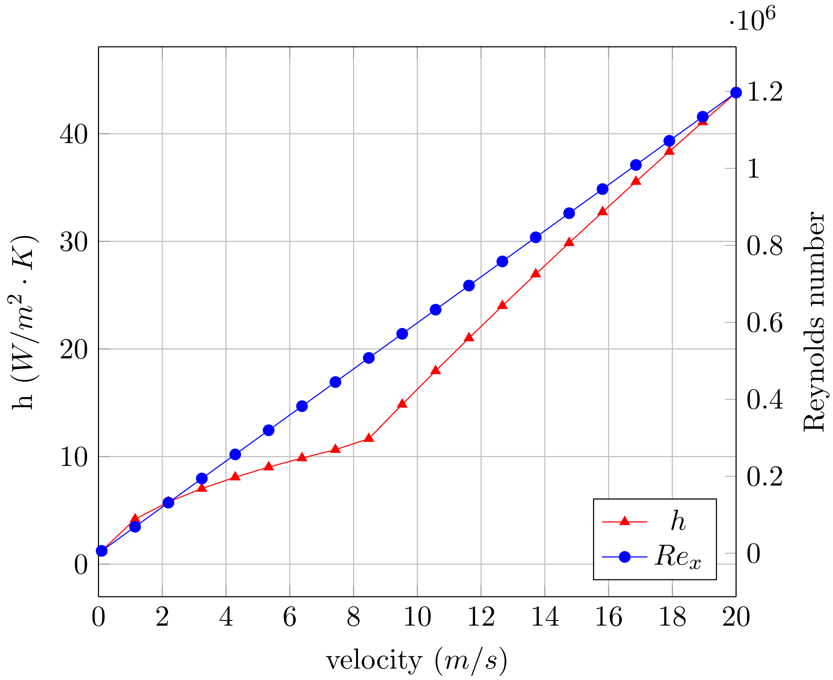


Figure 4.13: EFCplate, air

Example

Air is flowing over a flat plate that is 1.3 m long and 2 m wide. The free stream velocity is 7.5 m/sec. It is divided into three equal length surfaces for modeling purposes. The three surface nodes are labeled `1`, `2` and `3`, while the free stream air temperature is labeled `Tc`:

```
Begin Conductors
  ...
  surf1  EFCplate  1  Tc  air 7.5 0.0     0.43333  0.8666
  surf2  EFCplate  2  Tc  air 7.5 0.43333 0.86667  0.8667
  surf3  EFCplate  3  Tc  air 7.5 0.86667 1.3      0.8667
  ...
End Conductors
```

4.2.7 `EFCsphere` External Forced Convection Over a Sphere

```
!                    nd_i  nd_j
! label   type       Ts    Tinf   parameters
   (S)   EFCsphere  (S)    (S)    (S)  (R)  (R)     ! material, velocity, D
```

Type	Line Command		
Scope	`Conductors` block		
Parameters	material, u, D	**Default**	None
	u is the fluid velocity		
	D is the diameter of the sphere		

Description

External forced convection over a sphere:

$$Q_{ij} = hA(T_i - T_j) = hA(T_s - T_\infty) \tag{4.44}$$

The heat transfer coefficient, h, is evaluated using the correlation (see Equation (7.48), p. 444, in [BLID11] and [Whi72]):

$$\overline{Nu}_D = 2 + \left(0.4Re_D^{1/2} + 0.06Re_D^{2/3}\right) Pr^{0.4} \left(\frac{\mu}{\mu_s}\right)^{1/4} \tag{4.45}$$

where D is the diameter of the sphere (see Figure 4.14) and the Reynolds number is:

$$Re_D = \frac{\rho u D}{\mu} = \frac{uD}{\nu} \tag{4.46}$$

The fluid properties are evaluated at the free stream fluid temperature, T_∞, except the viscosity, μ_s, is evaluated at the surface temperature, T_s. Once the Nusselt number is determined, the convection coefficient, h, is given by:

$$h = \frac{\overline{Nu}_D k}{D} \tag{4.47}$$

where k is the thermal conductivity of the fluid.

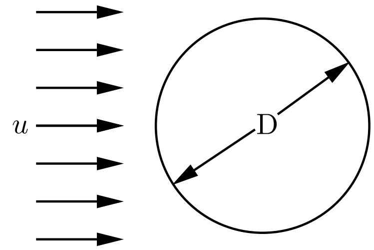


Figure 4.14: Geometry for Forced Convection over a Sphere

Example

A conductor is required for forced convection heat transfer from a sphere in cross-flow. The sphere diameter is 0.05 m. The fluid is water, with a velocity of 2 m/s. The conductor is given the label `fluid`, with a surface node called `wall` and the free stream temperature is call `Tinf`:

```
Begin Conductors
  ...
! label   type     nd_i  nd_j  parameters
  fluid  EFCsphere wall  Tinf  water  2.0  0.05  ! material, velocity, D
  ...
End Conductors
```

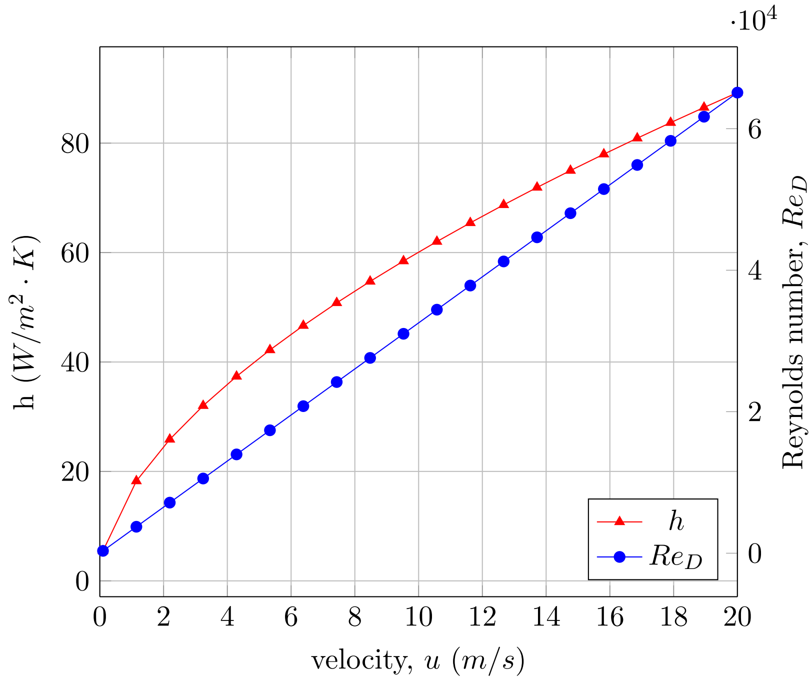


Figure 4.15: EFCsphere, air, $T_s = 50\ C$, $T_\infty = 25\ C$, $D = 0.0508\ m$

4.2.8 `INCvenc` Internal Natural Convection for a Vertical Rectangular Enclosure

```
! label   type        nd_i   nd_j   parameters
    (S)   INCvenc     (S)    (S)   (S)  (R)  (R)       ! material, W, H, A
```

Type Line Command
Scope `Conductors` block
Parameters material, W, H, A **Default** None
W is the width of the enclosure
H is the height of the enclosure
A is the surface area for convection

Description

Internal natural convection for a vertical rectangular enclosure:

$$Q_{ij} = hA(T_i - T_j) \tag{4.48}$$

The heat transfer coefficient, h, for a rectangular enclosure with aspect ratio $1 \leq W/H \leq 2$ and $10^3 \leq (Ra_W Pr)/(0.2 + Pr)$, is evaluated using the correlation (see Equation (9.51), p. 593 in [BLID11] or [Cat78]):

$$\overline{Nu}_W = 0.18 \left(\frac{Pr}{0.2 + Pr} Ra_W \right)^{0.29} \tag{4.49}$$

for aspect ratio $2 \leq W/H \leq 10$ and $10^3 \leq Ra_W \leq 10^{10}$, is evaluated using the correlation (see Equation (9.50), p. 591 in [BLID11] or [Cat78]):

$$\overline{Nu}_W = 0.22 \left(\frac{Pr}{0.2 + Pr} Ra_W \right)^{0.28} \left(\frac{H}{W} \right)^{-1/4} \tag{4.50}$$

For aspect ratio $10 \leq W/H \leq 40$ and $10^4 \leq Ra_W \leq 10^7$, is evaluated using the correlation (see Equation (9.52), p. 591 in [BLID11] or [ME69]):

$$\overline{Nu}_W = 0.42 Ra_W^{1/4} Pr^{0.012} \left(\frac{H}{W} \right)^{-0.3} \tag{4.51}$$

where W is the width of the enclosure (see Figure 4.16) and the Rayleigh number, Ra_W, is:

$$Ra_W = Gr Pr = \frac{g \rho^2 c \beta W^3 (T_s - T_\infty)}{k \mu} = \frac{g \beta W^3 (T_s - T_\infty)}{\nu \alpha} \tag{4.52}$$

Note that the fluid properties are evaluated at the average temperature, T_f:

$$T_f = \frac{T_1 + T_2}{2} \tag{4.53}$$

Once the Nusselt number is determined, the convection coefficient, h, is given by:

$$h = \frac{\overline{Nu}_W k}{W} \tag{4.54}$$

where k is the thermal conductivity of the fluid. The natural convection heat transfer coefficient

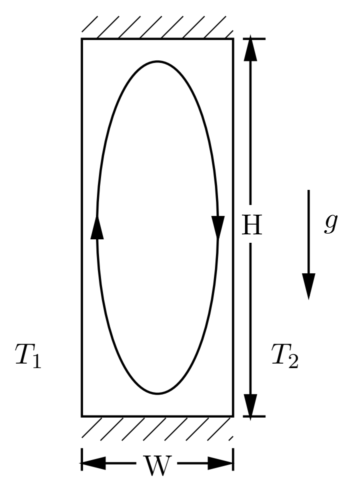


Figure 4.16: Geometry for Natural Convection in a Vertical Rectangular Enclosure

for air in a $1x1$ m enclosure, is shown in Figure 4.31 for a range of temperature differences.

Example

Natural convection in a vertical, rectangular enclosure that is 0.1 m wide and 3 m tall is modeled with a conductor named `panel`. The surface nodes are labeled `Tleft` and `Tright`. The material is air and the surface area for convection is 9 m^2:

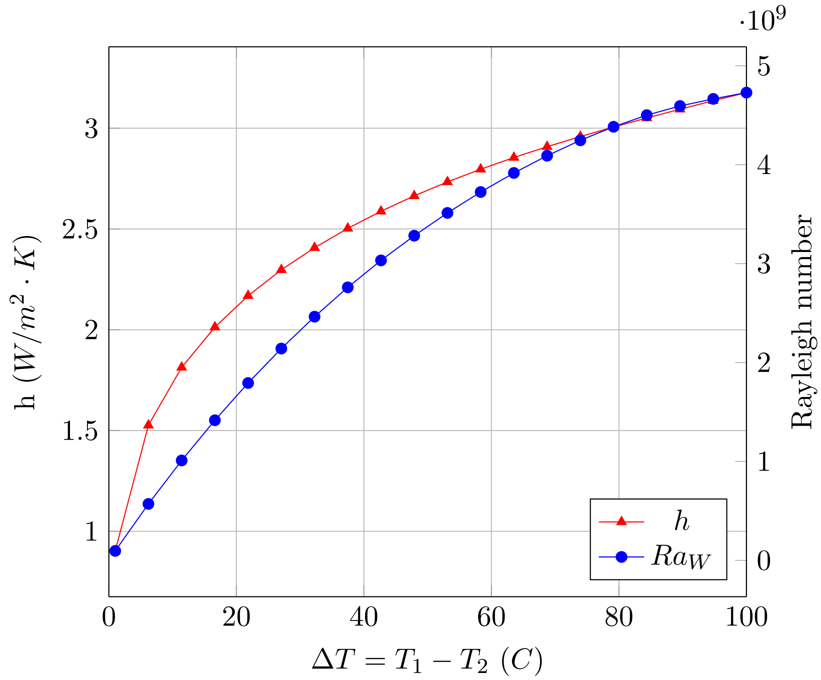


Figure 4.17: INCvenc, air, $T_1 = 25$ C, $W = 1$ m, $H = 1$ m

```
Begin Conductors
  ...
! label   type      nd_i   nd_j    parameters
  panel  INCvenc   Tleft  Tright  air 0.1 3.0 9.0 ! material, W, H, A
  ...
End Conductors
```

4.2.9 `ENChcyl` External Natural Convection Over a Horizontal Cylinder

```
! label    type        nd_i  nd_j  parameters
   (S)   ENChcyl       (S)   (S)   (S) (R) (R)     ! material, D, A
```

Type Line Command
Scope `Conductors` block
Parameters material, D, A **Default** None
D is the cylinder diamter
$A = \pi D \times L$ is the surface area for convection

Description

External natural convection over a horizontal cylinder:

$$Q_{ij} = hA(T_i - T_j) \tag{4.55}$$

The heat transfer coefficient, h, is evaluated using the correlation (see Equation (9.34), p. 581 in [BLID11]):

$$\overline{Nu}_D = \frac{\bar{h}D}{k} = \left\{ 0.60 + \frac{0.387 Ra_D^{1/6}}{\left[1 + (0.559/Pr)^{9/16}\right]^{8/27}} \right\}^2 \tag{4.56}$$

valid for $Ra_D \leq 10^{12}, Pr \geq 0.7$, where D is the diameter of the cylinder and the Rayleigh number, Ra, is:

$$Ra_D = GrPr = \frac{g\rho^2 c\beta D^3 (T_s - T_\infty)}{k\mu} = \frac{g\beta D^3 (T_s - T_\infty)}{\nu\alpha} \tag{4.57}$$

Note that the fluid properties are evaluated at the film temperature, T_f:

$$T_f = \frac{T_s + T_\infty}{2} \tag{4.58}$$

Once the Nusselt number is determined, the convection coefficient, h, is given by:

$$h = \frac{\overline{Nu}_D k}{D} \tag{4.59}$$

where k is the thermal conductivity of the fluid. The natural convection heat transfer coefficient for

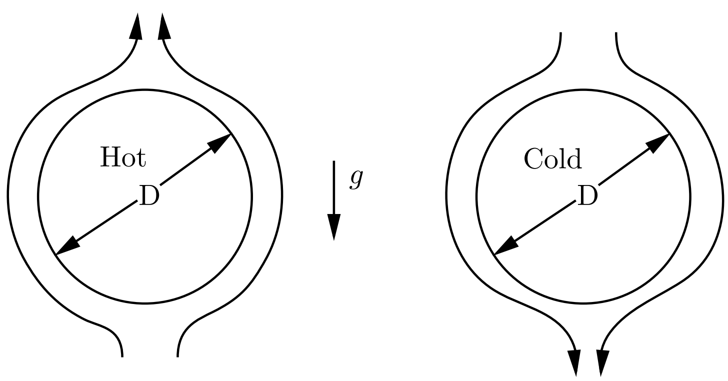


Figure 4.18: Geometry for Natural Convection from a Horizontal Cylinder

a 1/2 inch diameter horizontal cylinder, in air, is shown in Figure 4.19 for a range of temperature differences.

Example

A horizontal cylinder in air, with a diameter, $D = 0.05\ m$, and length of $L = 0.5\ m$ is modeled using a conductor labeled `101`. The surface node is labeled `25` and the free stream air node is labeled `Tc`. The surface area for convection is $A = \pi \times 0.05 \times 0.5 = 0.07854\ m^2$:

```
Begin Conductors
  ...
! label  type    nd_i  nd_j  parameters
   101  ENChcyl   25    Tc    air 0.05  0.07854    ! material, D, A
  ...
End Conductors
```

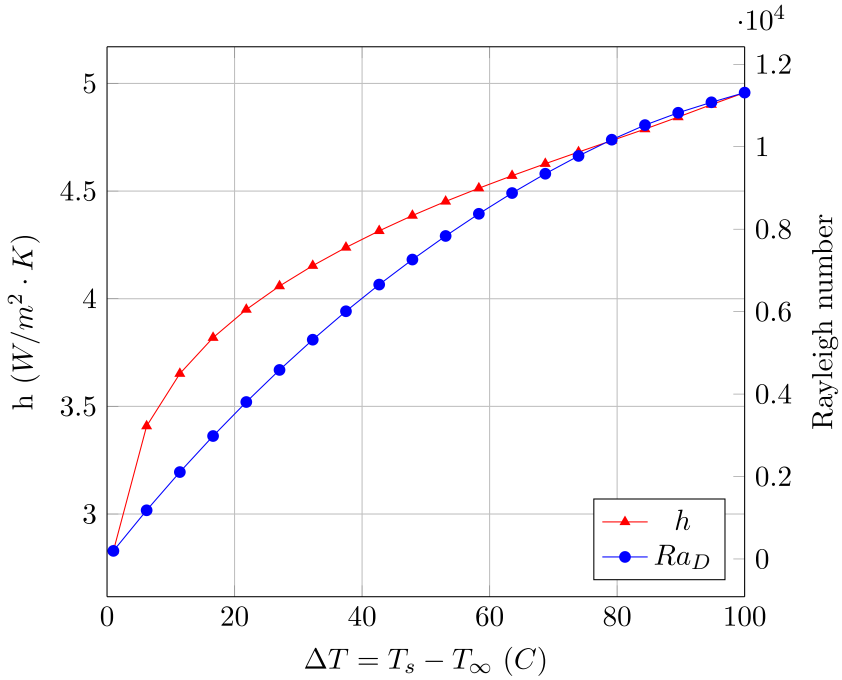


Figure 4.19: ENChcyl, air, $T_\infty = 25\ C$, $D = 0.0127\ m$

4.2.10 `ENChplatedown` External Natural Convection From a Downward Facing Horizontal Flat Plate

```
! label     type           nd_i   nd_j   parameters
    (S)   ENChplatedown    (S)    (S)   (S)  (R)  (R)       ! material, L, A
```

Type	Line Command
Scope	`Conductors` block
Parameters	material, L, A **Default** None
	L = Area/Perimeter = A/P is the characteristic length
	A = is the surface area for convection

Description

External natural convection from a downward facing, horizontal flat plate:

$$Q_{ij} = hA(T_i - T_j) \tag{4.60}$$

The heat transfer coefficient for laminar flow, $10^4 \lesssim Ra_L \lesssim 10^7$, from a cold plate is (see Equation (9.30), p. 578, in [BLID11]):

$$\overline{Nu}_L = 0.54 Ra_L^{1/4} \tag{4.61}$$

and for turbulent flow, $10^7 \lesssim Ra_L \lesssim 10^{11}$, from a hot plate is (see Equation (9.31), p. 578, in [BLID11]):

$$\overline{Nu}_L = 0.15 Ra_L^{1/3} \tag{4.62}$$

For a hot plate, $10^4 \lesssim Ra_L \lesssim 10^9$, the correlation is (see Equation (9.32), p. 578, in [BLID11]):

$$\overline{Nu}_L = 0.52 Ra_L^{1/5} \tag{4.63}$$

For a rectangular plate with width W and length L, the characteristic length is:

$$L = \frac{A}{P} = \frac{W \times L}{2(W + L)} \tag{4.64}$$

For a circular plate with radius R, the characteristic length is:

$$L = \frac{A}{P} = \frac{\pi R^2}{2\pi R} = \frac{R}{2} \tag{4.65}$$

Then the Rayleigh number, Ra_L, is:

$$Ra_L = GrPr = \frac{g\rho^2 c\beta L^3 (T_s - T_\infty)}{k\mu} = \frac{g\beta L^3 (T_s - T_\infty)}{\nu\alpha} \tag{4.66}$$

Note that the fluid properties are evaluated at the film temperature, T_f:

$$T_f = \frac{T_s + T_\infty}{2} \tag{4.67}$$

Once the Nusselt number is determined, the convection coefficient, h, is given by:

$$h = \frac{\overline{Nu}_L k}{L} \tag{4.68}$$

where k is the thermal conductivity of the fluid. The natural convection heat transfer coefficient

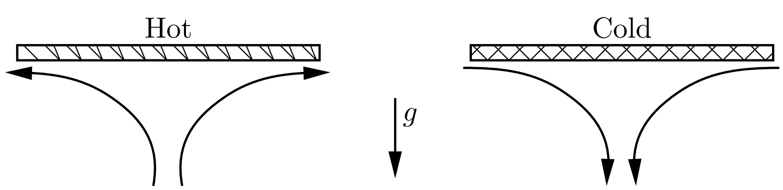


Figure 4.20: Geometry for Natural Convection from a Downward Facing Horizontal Flat Plate

for a $W = L = 0.6$ m square, downward facing, hot horizontal plate, in air, is shown in Figure 4.21 for a range of temperature differences.

Example

The natural convection from the bottom side of a horizontal circular disk in water, $R = 0.15$ m is modeled with a conductor labeled `1001`. The surface node is labeled `disk` and the free stream fluid node is labeled `Tc`. The characteristic length is $L = R/2 = 0.075$ m and the area is $A = \pi R^2 = 0.07069$ m^2:

```
Begin Conductors
  ...
! label    type        nd_i  nd_j  parameters
  1001   ENChplatedown disk   Tc   water 0.075 0.07069   ! material, L, A
  ...
End Conductors
```

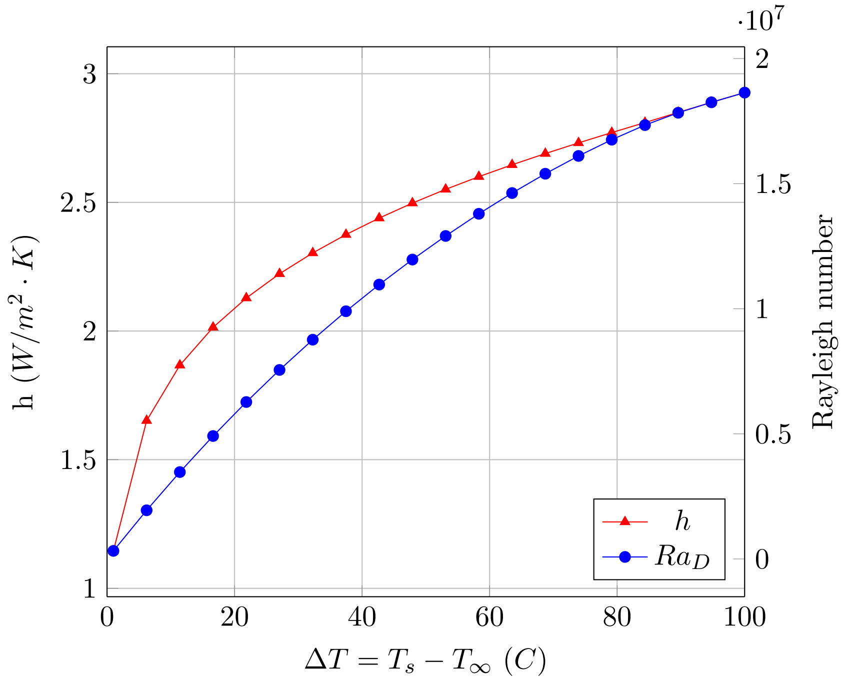


Figure 4.21: Hot ENChplatedown, air, $T_\infty = 25\ C$, $L = 0.15\ m$

4.2.11 `ENChplateup` External Natural Convection From an Upward Facing Horizontal Flat Plate

```
! label     type        nd_i   nd_j   parameters
   (S)   ENChplateup    (S)    (S)   (S)  (R)  (R)      ! material, L, A
```

Type Line Command
Scope `Conductors` block
Parameters material, L, A **Default** None
L = Area/Perimeter = A/P is the characteristic length
A = is the surface area for convection

Description

External natural convection from an upward facing, horizontal flat plate:

$$Q_{ij} = hA(T_i - T_j) \tag{4.69}$$

The heat transfer coefficient for laminar flow, $10^4 \lesssim Ra_L \lesssim 10^7$, from a hot plate is (see Equation (9.30), p. 578, in [BLID11]):

$$\overline{Nu}_L = 0.54 Ra_L^{1/4} \tag{4.70}$$

and for turbulent flow, $10^7 \lesssim Ra_L \lesssim 10^{11}$, from a hot plate is (see Equation (9.31), p. 578, in [BLID11]):

$$\overline{Nu}_L = 0.15 Ra_L^{1/3} \tag{4.71}$$

For a cold plate, $10^4 \lesssim Ra_L \lesssim 10^9$, the correlation is (see Equation (9.32), p. 578, in [BLID11]):

$$\overline{Nu}_L = 0.52 Ra_L^{1/5} \tag{4.72}$$

For a rectangular plate with width W and length L, the characteristic length is:

$$L = \frac{A}{P} = \frac{W \times L}{2(W + L)} \tag{4.73}$$

For a circular plate with radius R, the characteristic length is:

$$L = \frac{A}{P} = \frac{\pi R^2}{2\pi R} = \frac{R}{2} \tag{4.74}$$

Then the Rayleigh number, Ra_L, is:

$$Ra_L = Gr Pr = \frac{g \rho^2 c \beta L^3 (T_s - T_\infty)}{k \mu} = \frac{g \beta L^3 (T_s - T_\infty)}{\nu \alpha} \tag{4.75}$$

Note that the fluid properties are evaluated at the film temperature, T_f:

$$T_f = \frac{T_s + T_\infty}{2} \tag{4.76}$$

Once the Nusselt number is determined, the convection coefficient, h, is given by:

$$h = \frac{\overline{Nu}_L k}{L} \tag{4.77}$$

where k is the thermal conductivity of the fluid. The natural convection heat transfer coefficient

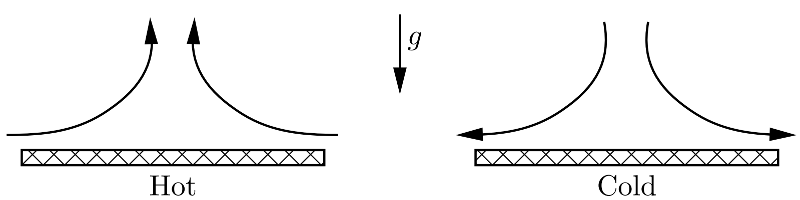


Figure 4.22: Geometry for Natural Convection from an Upward Facing Horizontal Flat Plate

for a $W = L = 0.6$ m square, upward facing, hot horizontal plate, in air, is shown in Figure 4.23 for a range of temperature differences. Note the transition from laminar to turbulent at a Rayleigh number of 10^7.

Example

The natural convection from the upper side of a horizontal circular disk in water, $R = 0.15$ m is modeled with a conductor labeled `1001`. The surface node is labeled `disk` and the free stream fluid node is labeled `Tc`. The characteristic length is $L = R/2 = 0.075$ m and the area is $A = \pi R^2 = 0.07069$ m^2:

```
Begin Conductors
  ...
! label    type        nd_i  nd_j  parameters
  1001  ENChplateup   disk   Tc  water 0.075 0.07069   ! material, L, A
  ...
End Conductors
```

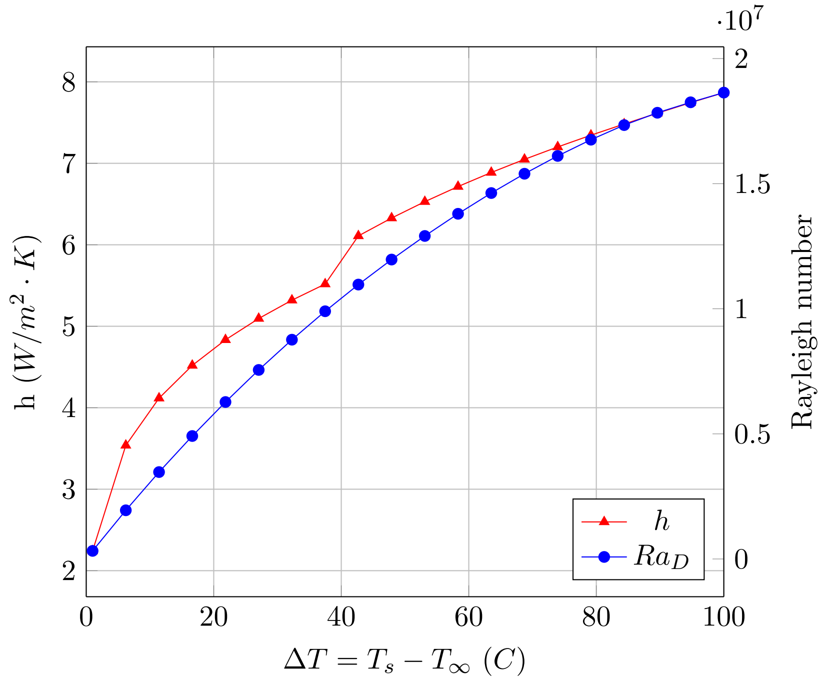


Figure 4.23: Hot ENChplateup, air, $T_\infty = 25\ C$, $L = 0.15\ m$

4.2.12 `ENCiplatedown` External Natural Convection From a Downward Facing Inclined Flat Plate

```
! label    type          nd_i  nd_j  parameters
  (S)  ENCiplatedown (S)   (S)   (S) (R) (R) (R) ! material, H, L=A/P, angle, A
```

Type Line Command
Scope `Conductors` block
Parameters material, H, $L = A/P$, θ, A **Default** None
H is the characteristic height
L is the characteristic length, L = area/perimeter = A/P
θ is the angle from vertical
A is the surface area for convection

Description

External natural convection from an downward facing, inclined flat plate:

$$Q_{ij} = hA(T_i - T_j) \tag{4.78}$$

For the stable case, an inclined hot plate facing down, $T_s > T_\infty$, the Nusselt number correlation for laminar flow from a vertical plate, $Ra_L \leq 10^9$, (see Equation (9.27), p. 573 in [BLID11] or [CC75]):

$$\overline{Nu}_H = 0.68 + \frac{0.670 Ra_H^{1/4}}{\left[1 + (0.492/\Pr)^{9/16}\right]^{4/9}} \tag{4.79}$$

and for turbulent flow, $Ra > 10^9$, the correlation used is (see Equation (9.26), p. 573 in [BLID11] or [CC75]):

$$\overline{Nu}_H = \left\{ 0.825 + \frac{0.387 Ra_H^{1/6}}{\left[1 + (0.492/Pr)^{9/16}\right]^{8/27}} \right\}^2 \tag{4.80}$$

where H is the length of the inclined plate and the Rayleigh number, Ra_H, is:

$$Ra_H = GrPr = \frac{(g\cos\theta)\rho^2 c\beta H^3 (T_s - T_\infty)}{k\mu} = \frac{(g\cos\theta)\beta H^3 (T_s - T_\infty)}{\nu\alpha} \tag{4.81}$$

Note that gravity, g, has been multiplied by the cosine of the angle θ (see Figure 4.26). The fluid properties are evaluated at the film temperature, T_f:

$$T_f = \frac{T_s + T_\infty}{2} \tag{4.82}$$

For the unstable case, an inclined cold plate facing down, $T_s < T_\infty$, the heat transfer coefficient is determined using the approach of Raithby and Hollands [RH98]. In this approach the heat transfer coefficient is evaluated for both a vertical plate with $g\cos\theta$ (see the stable case) and a horizontal plate with $g\cos(90-\theta)$. The heat transfer coefficient for laminar flow, $10^4 \lesssim Ra_H \lesssim 10^7$, from a cold plate is (see Equation (9.30), p. 578, in [BLID11]):

$$\overline{Nu}_L = 0.54 Ra_L^{1/4} \tag{4.83}$$

and for turbulent flow, $10^7 \lesssim Ra_L \lesssim 10^{11}$, from a hot plate is (see Equation (9.31), p. 578, in [BLID11]):

$$\overline{Nu}_L = 0.15 Ra_L^{1/3} \tag{4.84}$$

where $L = A/P$ is the ratio of the area of the plate to its perimeter and the Rayleigh number, Ra_L, is:

$$Ra_L = GrPr = \frac{(g\cos(90-\theta))\rho^2 c\beta L^3 (T_s - T_\infty)}{k\mu} = \frac{(g\cos(90-\theta))\beta L^3 (T_s - T_\infty)}{\nu\alpha} \tag{4.85}$$

Note that gravity, g, has been multiplied by the cosine of the angle $(90-\theta)$ (see Figure 4.26). The fluid properties are evaluated at the film temperature, T_f:

$$T_f = \frac{T_s + T_\infty}{2} \tag{4.86}$$

Once the Nusselt number is determined, the convection coefficient, h, is given by:

$$h = \frac{\overline{Nu}_L k}{L} \tag{4.87}$$

where k is the thermal conductivity of the fluid. The maximum of the two convection coefficients (vertical plate and horizontal plate) is then chosen for the inclined cold plate down. The natural convection heat transfer coefficient for a $H = W = 2.0$ m square, downward facing, hot plate, in air, is shown in Figure 4.25 for a range of temperature differences.

Example

A rectangular plate, in air, is inclined at an angle of $\theta = 50°$. The heat transfer from the lower surface is modeled using a conductor labeled `3021`, with the surface node named `plate` and the free stream temperature `Tc`. The plate is 0.2 m long and .5 m wide, giving a surface area of $A = 0.1$ m^2 and a perimeter of $P = 1.4$ m. The characteristic length is $L = A/P = 0.071$:

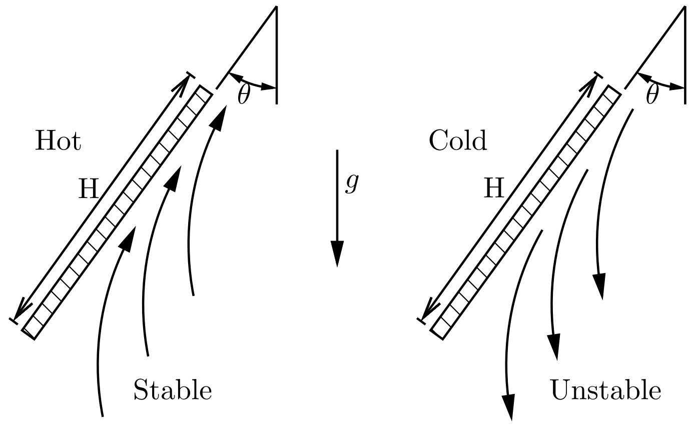


Figure 4.24: Geometry for Natural Convection from Downward Facing Inclined Flat Plate

```
Begin Conductors
  ...
! label    type       nd_i  nd_j  parameters
  3021 ENCiplatedown plate Tc air 0.2  0.071 50.0 0.1 ! material, H, L=A/P, angle, A
  ...
End Conductors
```

4.2.13 `ENCiplateup` External Natural Convection From an Upward Facing Inclined Flat Plate

```
! label    type       nd_i  nd_j  parameters
   (S)  ENCiplateup  (S)   (S)  (S) (R) (R) (R) (R)  ! material, H, L, angle, A
```

Type Line Command
Scope `Conductors` block
Parameters material, L, θ, A **Default** None
H is the length
L = area/perimeter = A/P is the characteristic length
θ is the angle from vertical
A is the surface area for convection

Description

External natural convection from an upward facing, inclined flat plate:

$$Q_{ij} = hA(T_i - T_j) \tag{4.88}$$

For the stable case, an inclined cold plate facing up, $T_s < T_\infty$, the Nusselt number correlation for

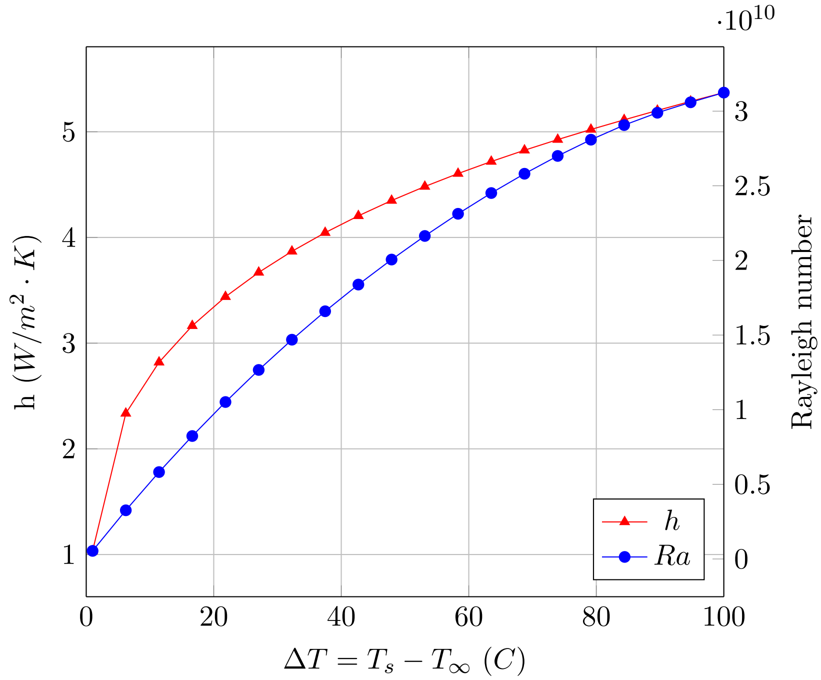


Figure 4.25: ENCiplatedown, air, $H = 2$ m, $L = A/P = 1/2$, $\theta = 45°$

laminar flow from a vertical plate, $Ra_L \leq 10^9$, (see Equation (9.27), p. 573 in [BLID11] or [CC75]):

$$\overline{Nu}_L = 0.68 + \frac{0.670Ra^{1/4}}{\left[1 + (0.492/\,\mathrm{Pr})^{9/16}\right]^{4/9}} \tag{4.89}$$

and for turbulent flow, $Ra > 10^9$, the correlation used is (see Equation (9.26), p. 573 in [BLID11] or [CC75]):

$$\overline{Nu}_L = \left\{0.825 + \frac{0.387Ra_L^{1/6}}{\left[1 + (0.492/Pr)^{9/16}\right]^{8/27}}\right\}^2 \tag{4.90}$$

where L is the length of the inclined plate and the Rayleigh number, Ra_L, is:

$$Ra_L = GrPr = \frac{(g\cos\theta)\rho^2 c\beta L^3\,(T_s - T_\infty)}{k\mu} = \frac{(g\cos\theta)\beta L^3\,(T_s - T_\infty)}{\nu\alpha} \tag{4.91}$$

Note that gravity, g, has been multiplied by the cosine of the angle θ (see Figure 4.26). The fluid properties are evaluated at the film temperature, T_f:

$$T_f = \frac{T_s + T_\infty}{2} \tag{4.92}$$

For the unstable case, an inclined hot plate facing up, $T_s > T_\infty$, the heat transfer coefficient is determined using the approach of Raithby and Hollands [RH98]. In this approach the heat transfer coefficient is evaluated for both a vertical plate with $g\cos\theta$ (see the stable case) and a horizontal plate with $g\cos(90-\theta)$. The heat transfer coefficient for laminar flow, $10^4 \lesssim Ra_L \lesssim 10^7$, from a hot plate is (see Equation (9.30), p. 578, in [BLID11]):

$$\overline{Nu}_L = 0.54Ra_L^{1/4} \tag{4.93}$$

and for turbulent flow, $10^7 \lesssim Ra_L \lesssim 10^{11}$, from a hot plate is (see Equation (9.31), p. 578, in [BLID11]):

$$\overline{Nu}_L = 0.15 Ra_L^{1/3} \tag{4.94}$$

where L is the length of the inclined plate and the Rayleigh number, Ra_L, is:

$$Ra_L = GrPr = \frac{(g\cos(90-\theta))\rho^2 c\beta L^3 (T_s - T_\infty)}{k\mu} = \frac{(g\cos(90-\theta))\beta L^3 (T_s - T_\infty)}{\nu\alpha} \tag{4.95}$$

Note that gravity, g, has been multiplied by the cosine of the angle $(90-\theta)$ (see Figure 4.26). The fluid properties are evaluated at the film temperature, T_f:

$$T_f = \frac{T_s + T_\infty}{2} \tag{4.96}$$

The maximum of the two Nusselt numbers (vertical plate and horizontal plate) is then chosen for the inclined hot plate up.

Once the Nusselt number is determined, the convection coefficient, h, is given by:

$$h = \frac{\overline{Nu}_L k}{L} \tag{4.97}$$

where k is the thermal conductivity of the fluid. The natural convection heat transfer coefficient

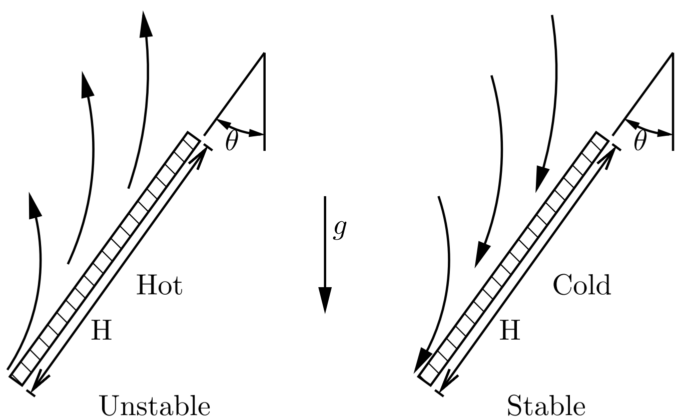


Figure 4.26: Geometry for Natural Convection from Upward Facing Inclined Flat Plate

for a $H = W = 2.0\ m$ square, upward facing hot plate, in air, is shown in Figure 4.27 for a range of temperature differences.

Example

A rectangular plate, in air, is inclined at an angle of $\theta = 50°$. The heat transfer from the upper surface is modeled using a conductor labeled `3021`, with the surface node named `plate` and the free stream temperature `Tc`. The plate is 0.2 m long and .5 m wide, giving a surface area of $A = 0.1\ m^2$ and a perimeter of $P = 1.4\ m$. The characteristic length is $L = A/P = 0.071$:

```
Begin Conductors
  ...
```

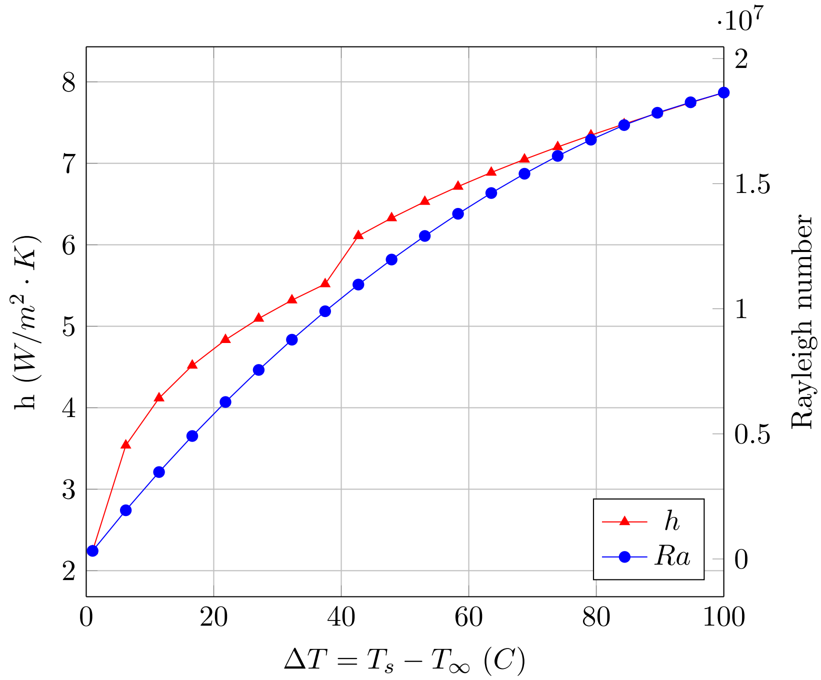


Figure 4.27: ENCiplateup, air, $H = 2$ m, $L = A/P = 1/2$, $\theta = 45°$

```
! label    type    nd_i nd_j  parameters
  3021 ENCiplateup plate Tc air 0.2 0.071 50.0 0.1 ! material, H, L=A/P, angle, A
  ...
End Conductors
```

4.2.14 `ENCsphere` External Natural Convection From a Sphere

```
! label   type      nd_i  nd_j  parameters
   (S)   ENCsphere (S)   (S)   (S)  (R)       ! material, D
```

Type Line Command
Scope `Conductors` block
Parameters material, D **Default** None

Description

External natural convection over a sphere, as shown in Figure 4.28, is:

$$Q_{ij} = hA(T_i - T_j) \tag{4.98}$$

The heat transfer coefficient, h, is evaluated using the correlation (Equation (9.35), p. 585, in [BLID11]):

$$\overline{Nu}_D = 2 + \frac{0.589 Ra_D^{1/4}}{\left[1 + (0.469/Pr)^{9/16}\right]^{4/9}} \tag{4.99}$$

valid for $Pr \gtrsim 0.7$ and $Ra_D \lesssim 10^{11}$. The Rayleigh number, Ra_D, is:

$$Ra_D = GrPr = \frac{g\rho^2 c\beta D^3 (T_s - T_\infty)}{k\mu} = \frac{g\beta D^3 (T_s - T_\infty)}{\nu\alpha} \tag{4.100}$$

Note that the fluid properties are evaluated at the film temperature, T_f:

$$T_f = \frac{T_s + T_\infty}{2} \tag{4.101}$$

Once the Nusselt number is determined, the convection coefficient, h, is given by:

$$h = \frac{\overline{Nu}_D k}{D} \tag{4.102}$$

where k is the thermal conductivity of the fluid. The heat transfer coefficient for a 2 inch diameter

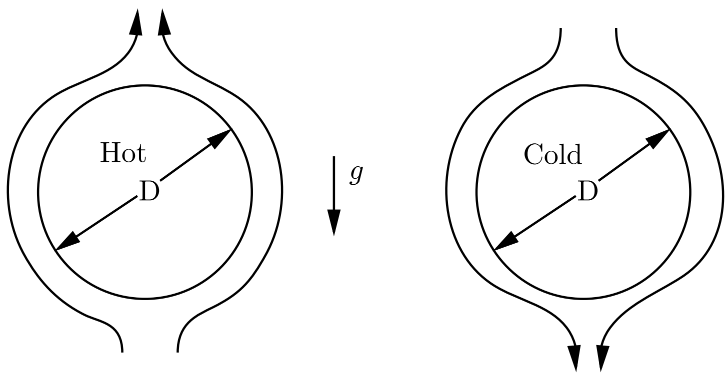


Figure 4.28: Geometry for External Natural Convection from a Sphere

sphere, in air, is shown in Figure 4.29 for a range of temperature differences.

Example

Natural convection from a sphere with a diameter of $D = 0.1\ m$, in water. The conductor is labeled `ball`, the surface node is labeled `1` and the free stream fluid temperature is labeled `Tc`:

```
Begin Conductors
   ...
! label   type     nd_i  nd_j  parameters
   ball  ENCsphere  1     Tc   water  0.1       ! material, D
   ...
End Conductors
```

4.2.15 `ENCvplate` External Natural Convection From a Vertical Flat Plate

```
! label    type       nd_i  nd_j  parameters
   (S)  ENCvplate     (S)   (S)  (S) (R) (R)      ! material, L, A
```

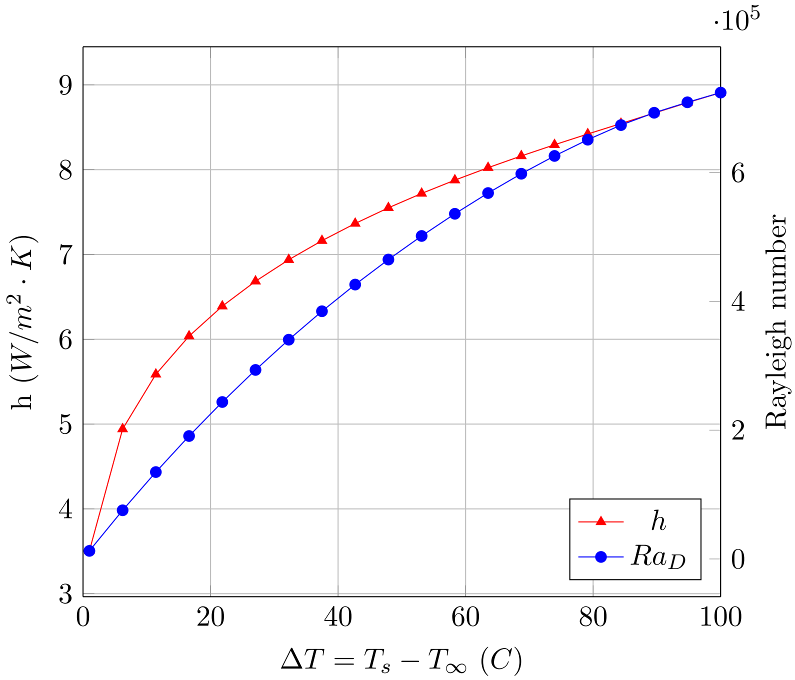


Figure 4.29: ENCsphere Convection Coefficient, air, $T_\infty = 25$ C, $D = 0.0508$ m

Type	Line Command		
Scope	`Conductors` block		
Parameters	material, L, A	**Default**	None
	L is the height of the plate		
	$A = L \times W$ is the surface area for convection		

Description

External natural convection from a vertical flat plate:

$$Q_{ij} = hA(T_i - T_j) \tag{4.103}$$

The heat transfer coefficient, h, for laminar flow, $Ra_L \lesssim 10^9$, is evaluated using the correlation (see Equation (9.27), p. 573 in [BLID11] or [CC75]):

$$\overline{Nu}_L = 0.68 + \frac{0.670 Ra_L^{1/4}}{\left[1 + (0.492/Pr)^{9/16}\right]^{4/9}} \tag{4.104}$$

For turbulent flow, $Ra_L > 10^9$, (see Equation (9.26), p. 573 in [BLID11] or [CC75]):

$$\overline{Nu}_L = \left\{ 0.825 + \frac{0.387 Ra_L^{1/6}}{\left[1 + (0.492/Pr)^{9/16}\right]^{8/27}} \right\}^2 \tag{4.105}$$

where L is the height of the vertical plate (see Figure 4.30) and the Rayleigh number, Ra_L, is:

$$Ra_L = Gr Pr = \frac{g \rho^2 c \beta L^3 (T_s - T_\infty)}{k \mu} = \frac{g \beta L^3 (T_s - T_\infty)}{\nu \alpha} \tag{4.106}$$

Note that the fluid properties are evaluated at the film temperature, T_f:

$$T_f = \frac{T_s + T_\infty}{2} \tag{4.107}$$

Once the Nusselt number is determined, the convection coefficient, h, is given by:

$$h = \frac{\overline{Nu}_L k}{L} \tag{4.108}$$

where k is the thermal conductivity of the fluid. The natural convection heat transfer coefficient for

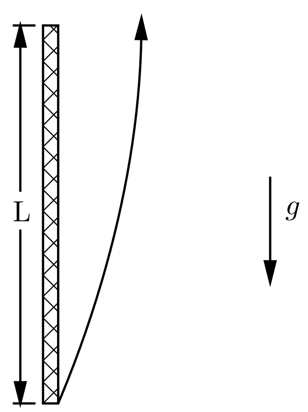


Figure 4.30: Geometry for Natural Convection from Vertical Flat Plate

a 48 inch tall vertical plate, in air, is shown in Figure 4.31 for a range of temperature differences.

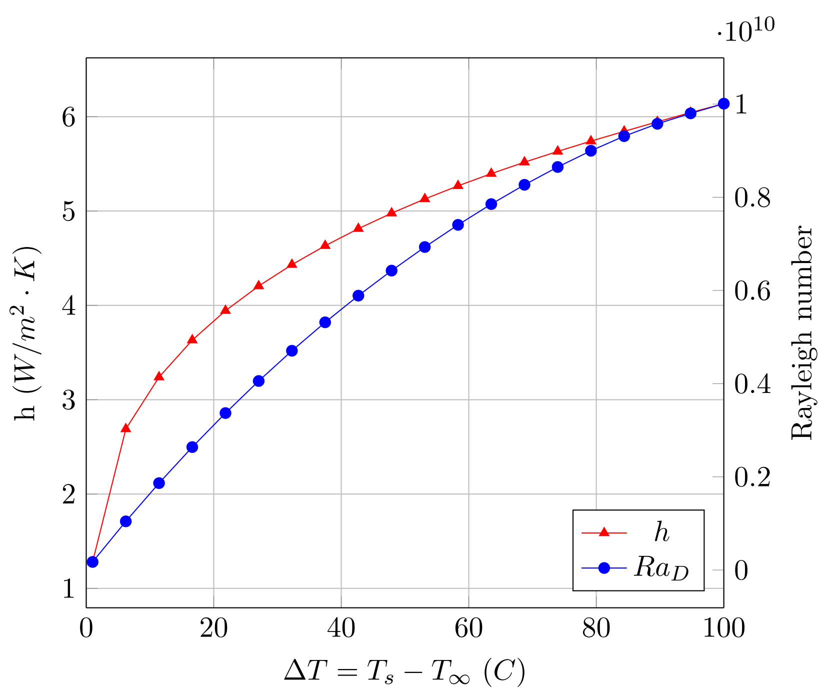


Figure 4.31: ENCvplate, air

Example

Natural convection from a vertical plate that is 3 m high by 2 m wide, in air, is modeled with a conductor labeled `room`, with the surface node labeled `wall` and the free stream air node labeled `Tc`. The surface ares for convection is $A = 3 \times 2 = 6\ m^2$:

```
Begin Conductors
  ...
! label   type      nd_i  nd_j  parameters
  room  ENCvplate   wall  Tc   air 3.0 6.0      ! material, L, A
  ...
End Conductors
```

4.2.16 `FCuser` Forced Convection User Function

```
! label  type  nd_i nd_j  parameters
   (S)  FCuser (S)  (S)  (S) (S) (R ...) (R) ! function, material, parameters, A
```

Type Line Command
Scope `Conductors` block
Parameters function, material, (R . . .), A **Default** None
function is the name of the user function
(R . . .) are the parameters passed to the function
A is the surface area for convection

Description

This convection conductor provides an interface to a user written function for the heat transfer coefficient for a forced convection scenario. The template is shown in Figure 4.32.

Example

4.2.17 `NCuser` Natural Convection User Function

```
! label  type  nd_i nd_j  parameters
   (S)  NCuser (S)  (S)  (S) (S) (R ...) (R) ! function, material, parameters, A
```

Type Line Command
Scope `Conductors` block
Parameters function, material, (R . . .), A **Default** None
function is the name of the user function
(R . . .) are the parameters passed to the function
A is the surface area for convection

```
function [h, Re, Nu] = funcname(mat, Ts, Tf, params)
%function [h, Re, Nu] = funcname(mat, Ts, Tf, params)
%
%  Description:
%
%    Template for a user written forced convection correlation
%    function.
%
%  Inputs:
%
%    mat      - material data structure for this conductor
%    Ts       - surface temperature
%    Tf       - fluid temperature
%    params() - vector of parameters parsed from the input file
%
%  Outputs:
%
%    h  - heat transfer coefficient
%    Re - Reynolds number
%    Nu - Nusselt number
%
%======================================================================

%  Evaluate the fluid properties

[k, rho, cp, mu, Pr] = fluidprop(mat, (Ts + Tf)/2.0);

%  Reynolds number

V = params(1);  % velocity parsed as parameter from the input file
D = params(2);  % characteristic length parsed from the input file
Re = (rho*V*D)/mu;

Nu =

h = (Nu*k)/D;
```

Figure 4.32: Template for Forced Convection User Function

Description

This convection conductor provides an interface to a user written function for the heat transfer coefficient for a natural convection scenario. The template is shown in Figure 4.33.

Example

4.3 Radiation

4.3.1 surfrad

```
! label    type       nd_i  nd_j  parameters
   (S)  surfrad       (S)   (S)  (R) (R)          ! emissivity, A
```

Type	Line Command
Scope	`Conductors` block
Parameters	ϵ, A **Default** None ϵ is the surface emissivity A is the surface area for radiation

Description

The heat transfer from a surface to a much larger environment, is given by:

$$Q_{ij} = \sigma \epsilon_i A_i (T_i^4 - T_j^4) \tag{4.109}$$

where σ is the Stefan-Boltzmann constant, ϵ_i is the surface emissivity and A_i is the surface area. Note that this relationship is only valid when $A_i \ll A_j$.

Example

A surface with emissivity of $\epsilon = 0.1$ and area $A = 2.3\ m^2$ is radiating to the night sky. The conductor is labeled `surf-sky` and the nodes are labeled `surf` and `sky`:

```
Begin Conductors
    ...
! label       type     nd_i  nd_j  parameters
  surf-sky  surfrad   surf  sky    0.1  2.3          ! emissivity, A
    ...
End Conductors
```

4.3.2 radiation

```
! label    type       nd_i  nd_j  parameters
   (S)  radiation     (S)   (S)  (R) (R)          ! script-F, A
```

```
function [h, Ra, Nu] = funcname(mat, Ts, Tf, params)
%function [h, Ra, Nu] = funcname(mat, Ts, Tf, params)
%
%  Description:
%
%    Template for a user written natural convection correlation
%    function.
%
%  Inputs:
%
%    mat      - material data structure for this conductor
%    Ts       - surface temperature
%    Tf       - fluid temperature
%    params() - vector of parameters parsed from the input file
%
%  Outputs:
%
%    h  - heat transfer coefficient
%    Ra - Rayleigh number
%    Nu - Nusselt number
%
%======================================================================

global g  %  Gravity

%  Evaluate the fluid properties

[k, rho, cp, mu, Pr] = fluidprop(mat, (Ts + Tf)/2.0);

[beta] = betaprop(mat, Tf);

%  Rayleigh number

L = params(1);  % characteristic length parsed from the input file
Ra = (g*rho^2*cp*beta*L^3*(abs(Ts - Tf)))/(k*mu);

Nu =

h = (Nu*k)/L;
```

Figure 4.33: Template for Natural Convection User Function

Type	Line Command		
Scope	`Conductors` block		
Parameters	$\mathscr{F}$, A	**Default**	None
	$\mathscr{F}$ is the exchange factor between the two surfaces		
	A is the surface area for radiation exchange		

Description

A radiation enclosure is modeled as N isothermal surfaces. The surfaces of the enclosure can have the following characteristics:

Diffuse: a modifier which means the property is not a function of direction.

Gray: a modifier indicating no dependence on wavelength.

Spectral: a modifier which means dependence on wavelength.

Specular: a modifier which means mirror-like reflection.

The heat transfer rate between two surfaces within the enclosure is:

$$Q_{ij} = \mathscr{F}_{ij} A_i \left(\sigma T_i^4 - \sigma T_j^4\right) \tag{4.110}$$

where $\mathscr{F}_{ij}$ is Hottel's over-all interchange factor (script-F or transfer factor) [Hot54, HS67]. It is a function of the emissivities of the two surfaces, ϵ_i and ϵ_j, and the geometric view factor between them, F_{ij}. Note symmetry:

$$\mathscr{F}_{ij} A_i = \mathscr{F}_{ji} A_j \tag{4.111}$$

and the row sum property:

$$\sum_{j=1}^{N} \mathscr{F}_{ij} = \epsilon_i \tag{4.112}$$

The temperature is linearized using a two term Taylor series expansion about the previous iteration temperature, T^*:

$$T_i^4 \approx (T_i^*)^4 + (T_i - T_i^*)\, 4(T_i^*)^3 \tag{4.113}$$

$$T_i^4 \approx 4(T_i^*)^3 T_i - 3(T_i^*)^4 \tag{4.114}$$

$$T_j^4 \approx \left(T_j^*\right)^4 + \left(T_j - T_j^*\right) 4\left(T_j^*\right)^3 \tag{4.115}$$

$$T_j^4 \approx 4\left(T_j^*\right)^3 T_j - 3\left(T_j^*\right)^4 \tag{4.116}$$

The linearized form of the heat transfer rate is:

$$Q_{ij} = \sigma \mathscr{F}_{ij} A_i \left[4(T_i^*)^3 T_i - 3(T_i^*)^4 - 4\left(T_j^*\right)^3 T_j + 3\left(T_j^*\right)^4\right] \tag{4.117}$$

or,

$$Q_{ij} = \sigma \mathscr{F}_{ij} A_i \left\lfloor \begin{array}{cc} 4(T_i^*)^3 & -4\left(T_j^*\right)^3 \end{array} \right\rfloor \left\{ \begin{array}{c} T_i \\ T_j \end{array} \right\} - \sigma \mathscr{F}_{ij} A_i \left\{3(T_i^*)^4 - 3\left(T_j^*\right)^4\right\} \tag{4.118}$$

<u>Relationship Between Interchange Factors and View Factors</u>

The interchange factor, $\mathscr{F}$, is the direct result of a Monte-Carlo ray tracing approach to evaluate enclosure radiation. This approach can include directional surface properties. If a method is used to evaluate the geometric view factors, F_{ij}, based on diffuse surface properties (commonly a hemicube/hemicube method), the $\mathscr{F}_{ij}$'s can be determined by the following method. See Chapter **??** for the generation of $\mathscr{F}$ and associated radiation conductors if all you have are the geometric view factors. The net heat flux using view factors, F_{ij}, is:

$$\{q\} = [\epsilon]([I] - [F][\rho])^{-1}([I] - [F])\{E_b\} \tag{4.119}$$

The net heat flux using exchange factors, $\mathscr{F}_{ij}$, is:

$$\{q\} = ([\epsilon] - [\mathscr{F}])\{E_b\} \tag{4.120}$$

The two enclosure heat fluxes are equal, so equating gives:

$$([\epsilon] - [\mathscr{F}])\{E_b\} = [\epsilon]([I] - [F][\rho])^{-1}([I] - [F])\{E_b\} \tag{4.121}$$

$$([\epsilon] - [\mathscr{F}]) = [\epsilon]([I] - [F][\rho])^{-1}([I] - [F]) \tag{4.122}$$

$$[\mathscr{F}] = [\epsilon]\left([I] - ([I] - [F][\rho])^{-1}([I] - [F])\right) \tag{4.123}$$

There is also the following relationship with Gebhart's absorption factors:

$$([I] - [F][\rho])[B] = [F][\epsilon] \tag{4.124}$$

Upon solving for the absorption factor matrix, $[B]$, the $\mathscr{F}$ are given by:

$$[\mathscr{F}] = [\epsilon][B] \tag{4.125}$$

where $[B]$ is a matrix of Gebhart's absorption factors [Geb71] (see [IB63], also see Example 5.17, page 228 in [HSM11]). Note that $[\rho]$ is a diagonal matrix of the surface reflectivities and $[\epsilon]$ is a diagonal matrix of the surface emissivities of the enclosure (for an opaque surface, $\rho + \epsilon = 1$, then $[\rho] = [I] - [\epsilon]$).

The transfer factor between two infinite plates is:

$$\mathscr{F}_{1\text{–}2} = \frac{1}{\frac{1}{\epsilon_1} + \frac{1}{\epsilon_2} - 1} = \frac{\epsilon_1\epsilon_2}{\epsilon_1 + \epsilon_2 - \epsilon_1\epsilon_2} \tag{4.126}$$

The transfer factor between a gray surface surrounded by another gray surface ($F_{1\text{–}2} = 1$ and $F_{2\text{–}1} \neq 0$, for example concentric cylinders (infinitely long) or spheres, see Equation (10.27), p. 554 in [LL12]) is:

$$\mathscr{F}_{1\text{–}2} = \frac{1}{\frac{1}{\epsilon_1} + \frac{A_1}{A_2}\left(\frac{1}{\epsilon_2} - 1\right)} \tag{4.127}$$

Example

Determine the radiation heat transfer between two concentric spheres. The area of a sphere is $A = 4\pi r^2$, leading to:

$$\mathscr{F}_{1-2} = \left[\frac{1}{\epsilon_1} + \frac{r_1^2}{r_2^2}\left(\frac{1}{\epsilon_2} - 1\right)\right]^{-1} \tag{4.128}$$

The inner sphere has a radius of $r_1 = 0.05\ m$ and an emissivity of $\epsilon = 0.1$. The outer sphere has a radius of $r_2 = 0.1\ m$ and an emissivity of $\epsilon = 0.9$. The exchange factor is:

$$\mathscr{F}_{1-2} = \left[\frac{1}{0.1} + \frac{0.05^2}{0.1^2}\left(\frac{1}{0.9} - 1\right)\right]^{-1} = 0.09972 \tag{4.129}$$

The surface area of the inner sphere is $A = 4\pi 0.05^2 = 0.03142\ m^2$. The radiation conductor from sphere 1 to sphere 2 is labeled `1-2` and the surface nodes are labeled `1` and `2`:

```
Begin Conductors
  ...
! label    type    nd_i nd_j  parameters
   1-2  radiation   1    2    0.09972  0.03142     ! script-F, A
  ...
End Conductors
```

4.4 Advection

4.4.1 advection

```
! label    type       nd_i  nd_j  parameters
   (S)  advection     (S)   (S)   (S) (R) (R)      ! material, velocity, A
```

Type	Line Command		
Scope	Conductors block		
Parameters	material, u, A	**Default**	None
	u is the average fluid velocity of the stream		
	A is the cross sectional area of the flow stream		

Description

The advection conductor is used to model the advective transport of energy, due to the motion of a fluid, in a flowing stream:

$$Q_{ij} = \dot{m} c_P \left(T_i - T_j\right) = \rho u A c_P \left(T_i - T_j\right) \tag{4.130}$$

where $\dot{m} = \rho u A$, is the mass flow rate of the fluid and c_P is the constant pressure specific heat.

Example

Water is flowing in a pipe with a diameter of $D = 0.05\ m$. The velocity is $u = 2\ m/s$. The cross sectional area of the flow stream is $A = \pi(0.05/2)^2 = 0.001963\ m^2$. The velocity is positive, so the upstream node is labeled 1 and the downstream node is labeled 2:

```
Begin Conductors
  ...
! label    type     nd_i  nd_j  parameters
   1001  advection   1     2    water 2.0 0.001963  ! material, velocity, A
  ...
End Conductors
```

4.4.2 outflow

```
! label   type        nd_i   nd_j   parameters
   (S)   outflow     (S)    (S)   (S)  (R)  (R)       ! material, velocity, A
```

Type Line Command
Scope `Conductors` block
Parameters material, u, A **Default** None
u is the average fluid velocity of the stream
A is the cross sectional area of the flow stream

Description

The outflow conductor is used to model the advective transport of energy, due to the motion of a fluid, in a flowing stream at the outflow boundary of the stream:

$$Q_{ij} = \dot{m}c_P\left(T_i - T_j\right) = \rho u A c_P\left(T_i - T_j\right) \tag{4.131}$$

where $\dot{m} = \rho u A$, is the mass flow rate of the fluid and c_P is the constant pressure specific heat.

Example

Water is flowing in a pipe with a diameter of $D = 0.05\ m$. The velocity is $u = 2\ m/s$. The cross sectional area of the flow stream is $A = \pi(0.05/2)^2 = 0.001963\ m^2$. The velocity is positive, so the upstream node is labeled `2` and the downstream outlet node is labeled `out`:

```
Begin Conductors
  ...
! label   type     nd_i  nd_j  parameters
   1001  outflow   2     out   water 2.0 0.001963  ! material, velocity, A
  ...
End Conductors
```

Chapter 5

Boundary Conditions

The boundary conditions block command is used to specify a variety of boundary conditions for the thermal model. Specified temperature and specified heat flux are currently defined.

```
Begin Boundary Conditions

! type       parameter  node(s)
  fixed_T    (R)        (S ...)  ! T
  heat_flux  (R) (R)    (S ...)  ! q, A

End Boundary Conditions
```

5.1 fixed_T

```
! type     parameter  node(s)
  fixed_T    (R)      (S ...)
```

Type	Line Command		
Scope	`Boundary Conditions` block		
Parameter	(R)	**Default**	None

Description

This line command will set a specified temperature boundary condition on one or more nodes in the thermal model.

Example

The free stream temperature for a convection correlation is set to 75.0 C:

```
  Begin Boundary Conditions
    ...
!   type     temperature  node(s)
    fixed_T      75.0     T_c  !  Free stream convection T
    ...
  End Boundary Conditions
```

5.2 heat_flux

```
! type        q   A   node(s)
  heat_flux  (R) (R)  (S ...)  ! q, A
```

Type Line Command
Scope Boundary Conditions block
Parameter (S ...) **Default** None

Description

This line command will apply a specified heat flux to one or more nodes in the thermal model.

Example

:

```
  Begin Boundary Conditions
    ...

    ...
  End Boundary Conditions
```

Chapter 6

Sources

The `Sources` command block is used to apply volumetric heating to the thermal model.

```
Begin Sources

! type    parameter(s)    node(s)
  qdot    (R)             (S ...)  ! q̇
  Qsrc    (R)             (S ...)  ! Q
  tstatQ  (R) (S) (R) (R) (S ...)  ! Q, thermostat node, Ton, Toff

End Sources
```

6.1 qdot

```
! type  q̇   node(s)
  qdot (R)  (S ...)
```

Type Line Command
Scope `Sources` block
Parameter $\dot{q}$ **Default** None

Description

This line command will specify a volumetric heat source per unit volume, $\dot{q}$, to the specified nodes. Note that the volume of the node must be defined in the Nodes block command.

Example

A volumetric heat source per unit volume of $10.3\,W/m^3$ is applied to the nodes labeled `100` and `101`:

```
Begin Sources
  ...
! type  qdot  nodes
  qdot  10.3  100  101
  ...
End Sources
```

6.2 Qsrc

```
! type   Q   node(s)
  Qsrc  (R)  (S ...)
```

Type	Line Command		
Scope	Sources block		
Parameter	Q	**Default**	None

Description

This line command will apply a total heat source, $Q = \dot{q}V$, to the specified node or nodes. Note that the node label has to exist either in a nodes block or a conductors block.

Example

Set the total heat source to $5\,W$ for the node labeled cpu:

```
Begin Sources
  ...
! type   Q   node
  Qsrc  5.0  cpu
  ...
End Sources
```

6.3 tstatQ

```
! type     Q  Tnode Ton Toff  node(s)
  tstatQ  (R)  (S)  (R) (R)   (S ...)  ! Q, thermostat node, Ton, Toff
```

Type	Line Command
Scope	`Sources` block
Parameter	Q `Tnode Ton Toff` **Default** None
	Q total heat source applied
	`Tnode` is the label of the thermostat node
	`Ton` source is on if node T is less than this T
	`Toff` source is off if node T is greater than this T

Description

This line command will apply a total heat source, $Q = \dot{q}V$, to the specified node or nodes, when the node temperature is between the two set point temperatures. Otherwise the heat source will be zero. The primary use for this source term is to model a heater with a thermostat control. Note that the node label has to exist either in a nodes block or a conductors block for both the thermostat and the source.

Example

A $1,000\,W$ electric heater is applied to the node labeled `plate`. The heater will turn on when the temperature of the node labeled `Tair` is below Ton and will turn off when it is above Toff:

```
Begin Sources
  ...
! type     Q      Tnode Ton   Toff  node
  tstatQ 1000.0  Tair  80.0  85.0  plate
  ...
End Sources
```

Chapter 7

Initial Conditions

The `Initial Conditions` command block is used to set the initial temperature of all and/or specified nodes in the model. By default, the initial temperature for both steady and transient analysis is zero.

```
Begin Initial Conditions

! Initial T  apply to all nodes in the model
     (R)     all

! Initial T  node(s)
     (R)     (S ...)

    Read Restart = (S)    ! restart file name

End Initial Conditions
```

Description

By default the initial temperature of all the nodes in the model is set to zero. By supplying an initial temperature in the `Initial Conditions` command block with the `all` keyword that is realistic to the problem at hand, the nonlinear behavior will be improved. Any specific node with a specified temperature value will override the `all` conditions.

Example

The initial temperature of all the nodes in the model is set to 20.0 C and the node labeled wall, is set to 5.0 C:

```
Begin Initial Conditions
!  T    node(s)
  20.0  all
   5.0  wall
End Initial Conditions
```

7.1 Read Restart

```
read restart = (S)  ! restart file name
```

Type	Line Command		
Scope	`Initial Conditions` block		
Parameter	(S)	**Default**	None

Description

A restart file is written at the end of every TNSolver run. The file contains a time stamp, followed by a list of single lines with the node label and its temperature.

```
Time = (R)
! label  temperature
   (S)      (R)
   (S)      (R)
   (S)      (R)
   (S)      (R)
   (S)      (R)
  ...
```

Figure 7.1: Restart File Format

Example

The initial temperatures are stored in the file named: `temperature.rst`.

```
Begin Initial Conditions
  ...
  read restart = temperatures.rst  ! restart file name
  ...
End Initial Conditions
```

Chapter 8

Functions

The `Functions` command block is used to define functions for use in the model. Currently functions of time are supported. A function can be used in replace of a parameter in the input file. The value of the parameter is then given by the function, based on the current value of the independent variable.

```
Begin Functions

  Begin Constant (S)
    (R)
  End Constant (S)

  Begin Time Table (S)
    (R)  (R)
    ...
    (R)  (R)
  End Time Table (S)

  Begin Time Spline (S)
    (R)  (R)
    ...
    (R)  (R)
  End Time Spline (S)

End Functions
```

Description

8.1 Constant

```
Begin Functions
  ...
  Begin Constant (S)  !  function name
    (R)   ! value
  End Constant (S)
  ...
End Functions
```

Type	Block Command		
Scope	`Functions` block		
Parameters	(S) (R)	**Default**	None
	(S) is the name of the function		
	(R) is the value of the constant function		

Description

Example

8.2 Time Table

```
Begin Functions
  ...
  Begin Time Table (S)  !  function name
  ! time value
     (R)  (R)
     ...
     (R)  (R)
  End Time Table (S)
  ...
End Functions
```

Type	Block Command		
Scope	`Functions` block		
Parameters	(S) (R) (R)	**Default**	None
	(S) is the name of the function		
	(R) is the time of the function		
	(R) is the value of the function		

Description

This function is a piecewise linear interpolation of the table of function values.

Example

8.3 Time Spline

```
Begin Functions
  ...
  Begin Time Spline (S)  !  function name
  ! time value
     (R)  (R)
     ...
     (R)  (R)
  End Time Spline (S)
  ...
End Functions
```

Type	Block Command		
Scope	`Functions` block		
Parameters	(S) (R) (R)	**Default**	None
	(S) is the name of the function		
	(R) is the time of the function		
	(R) is the value of the function		

Description

This function is a piecewise cubic Hermite interpolating polynomial interpolation of the table of function values.

Example

Chapter 9

Radiation Enclosure

The `Radiation Enclosure` command block will generate all the radiation conductors for an enclosure. There can be one or more enclosures in a thermal model.

```
Begin Radiation Enclosure

! label  emiss  area  view factor matrix entries
   (S)    (R)    (R)   (R ...)

End Radiation Enclosure
```

Description

A radiation enclosure is defined by the number of surfaces in the enclosure. Each surface is given a label and has an emissivity and area associated with it. Each surface will have a view factor to each of the other surfaces in the enclosure. Note that a view factor of zero must be included in the view factor matrix in order to have a square matrix on input. The radiation conductors that are generated for the enclosure are reported in the output file.

Example

The view factor between two surfaces forming a long groove, as shown in Figure 9.1, is given by:

$$F_{1-2} = F_{2-1} = 1 - sin\left(\frac{\alpha}{2}\right) \tag{9.1}$$

In addition we have reciprocity:

$$A_i F_{i-j} = A_j F_{j-i} \tag{9.2}$$

and row sum:

$$\sum_{i=1}^{N} F_{i-j} = 1 \tag{9.3}$$

For a value of $\alpha = 30^\circ$ and $w = 1.2$, the enclosure radiation problem is specified by:

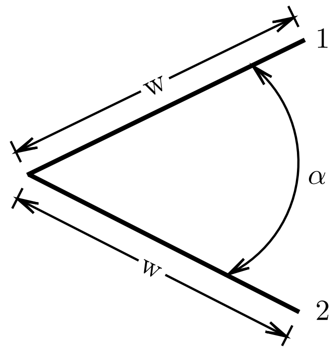


Figure 9.1: View Factor Example Geometry

```
Begin Radiation Enclosure
! label  emiss  area  view factor matrix entries
   1      0.3    1.2   0.0      0.7412   0.2588
   2      0.8    1.2   0.7412   0.0      0.2588
  env     1.0    2.0   0.15528  0.15528  0.68944
End Radiation Enclosure
```

The output file contains the generated conductors for the enclosure:

```
   Generated Radiation Conductors for this Enclosure

   Label        Type       Node i     Node j    script-F     Area
 ---------- ------------- ---------- ---------- ---------- ----------
        1-2     radiation          1          2    0.19271        1.2
      1-env     radiation          1        env  0.0965774        1.2
      2-env     radiation          2        env   0.340662        1.2
```

Chapter 10

Material Properties

Material properties are specified either by the built-in library or can be defined in the input file. Material properties defined in the `Material` command block in the input file will overide the material properties in the built-in library.

Table 10.1: Required Material Properties

property	steady	transient	forced convection	natural convection	advection
conductivity, k	X	X	X	X	
density, ρ		X	X	X	X
specific heat, c_v		X			
c_p				X	X
dynamic viscosity, μ			X	X	
thermal expansion, β				X	
Prandtl number, $Pr = c_p\mu/k$			X	X	

10.1 Material Library

TNSolver has a built in material library for use with models. If you are satisfied with these properties, then you can use the material name with any conductor in the model as shown in Table 10.2. Note that a user supplied material in the input file will override an equivalent named material in the library.

Table 10.2: Built-in Material Library Entries

material	Notes
`air`	At atmospheric pressure, 101.325 kPa
`water`	At atmospheric pressure, 101.325 kPa
`steel`	AISI 1010
`fir`	Perpendicular to the grain

10.2 Material

The Material command block is used to define material properties for use in the model. You must give a material a name that can then be used with any conductor in the model that requires a material. The material can be constant, piecewise linear, spline or a polynomial function of temperature. In addition a gas can be treated as an ideal gas, if desired.

```
Begin Material (S)   ! material name

  State = (S)  ! {gas|liquid|solid}, required for all materials

  Density = {(R)|ideal gas}  ! ideal gas: ρ = P/RT
  Density {Table|Spline}
!    T  density
    (R)  (R)
    ...
    (R)  (R)
  End Density {Table|Spline}
  Density Polynomial
    (R ...)          ! a0 a1 a2 ...
    range = (R) (R)  ! range begin and end
  End Density Polynomial

  Conductivity = (R)
  Conductivity {Table|Spline}
!    T    k
    (R)  (R)
    ...
    (R)  (R)
  End Conductivity {Table|Spline}
  Conductivity Polynomial
    (R ...)          ! a0 a1 a2 ...
    range = (R) (R)  ! range begin and end
  End Conductivity Polynomial

  {Specific Heat|c_v} = (R)  ! constant volume specific heat
  {Specific Heat|c_v} {Table|Spline}
!    T   c_v
    (R)  (R)
    ...
    (R)  (R)
  End {Specific Heat|c_v} {Table|Spline}
  {Specific Heat|c_v} Polynomial
    (R ...)          ! a0 a1 a2 ...
    range = (R) (R)  ! range begin and end
  End {Specific heat|c_v} Polynomial
```

```
  c_p = (R)  ! constant pressure specific heat
  c_p {Table|Spline}
!    T   c_p
    (R)  (R)
    ...
    (R)  (R)
  End c_p {Table|Spline}
  c_p Polynomial
    (R ...)          ! a0 a1 a2 ...
    range = (R) (R)  ! range begin and end
  End c_p Polynomial

  Viscosity = (R)   ! dynamic viscosity: μ
  Viscosity {Table|Spline}
!    T  viscosity, μ
    (R)  (R)
    ...
    (R)  (R)
  End Viscosity {Table|Spline}
  Viscosity Polynomial
    (R ...)          ! a0 a1 a2 ...
    range = (R) (R)  ! range begin and end
  End Viscosity Polynomial

  Beta = {(R)|ideal gas}  ! thermal expansion coefficient,
                          ! ideal gas: β = 1/T
  Beta {Table|Spline}
!    T   beta, β
    (R)  (R)
    ...
    (R)  (R)
  End Beta {Table|Spline}
  Beta Polynomial
    (R ...)          ! a0 a1 a2 ...
    range = (R) (R)  ! range begin and end
  End Beta Polynomial

  Pr = (R)                ! Prandtl number, Pr = c_p μ/k
  Pr {Table|Spline}
!    T   Pr, Pr = c_p μ/k
    (R)  (R)
    ...
    (R)  (R)
  End Pr {Table|Spline}
  Pr Polynomial
    (R ...)          ! a0 a1 a2 ...
```

```
    range = (R) (R)  ! range begin and end
  End Pr Polynomial

  gas constant = (R)  ! gas constant for use with ideal gas: R = R̂/M

  reference = (S ...)

End Material (S)
```

10.2.1 State

```
Begin Material (S)
  ...
  State = <solid|liquid|gas>
  ...
End Material (S)
```

Type	Line Command		
Scope	`Material` block		
Parameters	¡solid—liquid—gas¿	**Default**	None

Description

The state of the material must be defined as either solid, liquid or gas.

Example

10.2.2 Density =

```
Begin Material (S)
  ...
  Density = (R)
  ...
End Material (S)
```

Type	Line Command		
Scope	`Material` block		
Parameters	(R)	**Default**	None
	(R) is the density		

Description

A constant density is specified.

Example

10.2.3 `Density Table`

```
Begin Material (S)
  ...
  Density Table
!    T   rho
    (R)  (R)
  End Density Table
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R) (R) (R) is the temperature (R) is the conductivity	**Default**	None

Description

A piecewise linear interpolation of the thermal conductivity for the material as a function of temperature.

Example

10.2.4 `Density Spline`

```
Begin Material (S)
  ...
  Density Spline
!    T   rho
    (R)  (R)
  End Density Spline
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R) (R) (R) is the temperature (R) is the density	**Default**	None

Description

A monotonic spline interpolation of the density for the material as a function of temperature.

Example

10.2.5 `Density Polynomial`

```
Begin Material (S)
  ...
  Density Polynomial
    (R...)
    range = (R)  (R)
  End Density Polynomial
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R...)	**Default**	None
	(R...) are the coefficients of the polynomial		
	(R) (R) is the optional range of the polynomial		

Description

A polynomial is defined for the density of the material as a function of temperature. The range can be specified, if desired.

Example

10.2.6 `Conductivity =`

```
Begin Material (S)
  ...
  Conductivity = (R)
  ...
End Material (S)
```

Type	Line Command		
Scope	`Material` block		
Parameters	(R)	**Default**	None
	(R) is the conductivity		

Description

A constant conductivity is specified.

Example

10.2.7 `Conductivity Table`

```
Begin Material (S)
  ...
  Conductivity Table
!    T    k
    (R)  (R)
  End Conductivity Table
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R) (R) (R) is the temperature (R) is the conductivity	**Default**	None

Description

A piecewise linear interpolation of the thermal conductivity for the material as a function of temperature.

Example

10.2.8 `Conductivity Spline`

```
Begin Material (S)
  ...
  Conductivity Spline
!    T    k
    (R)  (R)
  End Conductivity Spline
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R) (R)	**Default**	None
	(R) is the temperature		
	(R) is the conductivity		

Description

A monotonic spline interpolation of the conductivity for the material as a function of temperature.

Example

10.2.9 `Conductivity Polynomial`

```
Begin Material (S)
  ...
  Conductivity Polynomial
    (R...)
    range = (R)  (R)
  End Conductivity Polynomial
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R...)	**Default**	None
	(R...) are the coefficients of the polynomial		
	(R) (R) is the optional range of the polynomial		

Description

A polynomial is defined for the conductivity of the material as a function of temperature. The range can be specified, if desired.

Example

10.2.10 `{Specific Heat|c_v} =`

```
Begin Material (S)
  ...
  {Specific Heat|c_v} = (R)
  ...
End Material (S)
```

Type	Line Command		
Scope	`Material` block		
Parameters	(R)	**Default**	None
	(R) is the specific heat		

Description

A constant specific heat is specified.

Example

10.2.11 {Specific Heat|c_v} Table

```
Begin Material (S)
  ...
  {Specific Heat|c_v} Table
!    T   c_v
    (R)  (R)
  End {Specific Heat|c_v} Table
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R) (R)	**Default**	None
	(R) is the temperature		
	(R) is the specific heat		

Description

A piecewise linear interpolation of the specific heat for the material as a function of temperature.

Example

10.2.12 {Specific Heat|c_v} Spline

```
Begin Material (S)
  ...
  {Specific Heat|c_v} Spline
!    T   c_v
    (R)  (R)
  End {Specific Heat|c_v} Spline
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R) (R)	**Default**	None
	(R) is the temperature		
	(R) is the specific heat		

Description

A monotonic spline interpolation of the specific heat for the material as a function of temperature.

Example

10.2.13 {Specific Heat|c_v} Polynomial

```
Begin Material (S)
  ...
  {Specific Heat|c_v} Polynomial
    (R...)
    range = (R)  (R)
  End {Specific Heat|c_v} Polynomial
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R...)	**Default**	None
	(R...) are the coefficients of the polynomial		
	(R) (R) is the optional range of the polynomial		

Description

A polynomial is defined for the specific heat of the material as a function of temperature. The range can be specified, if desired.

Example

10.2.14 c_p =

```
Begin Material (S)
  ...
  c_p = (R)
  ...
End Material (S)
```

Type	Line Command		
Scope	`Material` block		
Parameters	(R)	**Default**	None
	(R) is the c_p		

Description

A constant c_p is specified.

Example

10.2.15 `c_p Table`

```
Begin Material (S)
  ...
  c_p Table
!     T    c_p
    (R)  (R)
  End c_p Table
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R) (R)	**Default**	None
	(R) is the temperature		
	(R) is the c_p		

Description

A piecewise linear interpolation of the c_p for the material as a function of temperature.

Example

10.2.16 `c_p Spline`

```
Begin Material (S)
  ...
  c_p Spline
!     T    c_p
    (R)  (R)
  End c_p Spline
  ...
End Material (S)
```

Type Block Command
Scope `Material` block
Parameters (R) (R) **Default** None
(R) is the temperature
(R) is the c_p

Description

A monotonic spline interpolation of the c_p for the material as a function of temperature.

Example

10.2.17 `c_p Polynomial`

```
Begin Material (S)
  ...
  c_p Polynomial
    (R...)
    range = (R)  (R)
  End c_p Polynomial
  ...
End Material (S)
```

Type Block Command
Scope `Material` block
Parameters (R...) **Default** None
(R...) are the coefficients of the polynomial
(R) (R) is the optional range of the polynomial

Description

A polynomial is defined for the c_p of the material as a function of temperature. The range can be specified, if desired.

Example

10.2.18 `Viscosity =`

```
Begin Material (S)
  ...
  Viscosity = (R)
  ...
End Material (S)
```

Type	Line Command		
Scope	`Material` block		
Parameters	(R)	**Default**	None
	(R) is the viscosity		

Description

A constant viscosity is specified.

Example

10.2.19 `Viscosity Table`

```
Begin Material (S)
  ...
  Viscosity Table
!    T  viscosity
    (R)  (R)
  End Viscosity Table
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R) (R)	**Default**	None
	(R) is the temperature		
	(R) is the viscosity		

Description

A piecewise linear interpolation of the viscosity for the material as a function of temperature.

Example

10.2.20 `Viscosity Spline`

```
Begin Material (S)
  ...
  Viscosity Spline
!    T  viscosity
    (R)  (R)
  End Viscosity Spline
  ...
End Material (S)
```

Type	Block Command
Scope	`Material` block
Parameters	(R) (R) **Default** None
	(R) is the temperature
	(R) is the viscosity

Description

A monotonic spline interpolation of the viscosity for the material as a function of temperature.

Example

10.2.21 `Viscosity Polynomial`

```
Begin Material (S)
  ...
  Viscosity Polynomial
    (R...)
    range = (R)  (R)
  End Viscosity Polynomial
  ...
End Material (S)
```

Type	Block Command
Scope	`Material` block
Parameters	(R...) **Default** None
	(R...) are the coefficients of the polynomial
	(R) (R) is the optional range of the polynomial

Description

A polynomial is defined for the viscosity of the material as a function of temperature. The range can be specified, if desired.

Example

10.2.22 `Beta =`

```
Begin Material (S)
  ...
  Beta = {(R)|ideal gas}
  ...
End Material (S)
```

Type	Line Command		
Scope	`Material` block		
Parameters	{(R)—ideal gas}	**Default**	None
	(R) is the thermal expansion coefficient, β		

Description

A constant thermal expansion coefficient, β, is specified.

Example

10.2.23 `Beta Table`

```
Begin Material (S)
  ...
  Beta Table
!    T  beta
    (R)  (R)
  End Beta Table
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R) (R)	**Default**	None
	(R) is the temperature		
	(R) is the thermal expansion coefficient, β		

Description

A piecewise linear interpolation of the thermal expansion coefficient, β, for the material as a function of temperature.

Example

10.2.24 `Beta Spline`

```
Begin Material (S)
  ...
  Beta Spline
!    T  beta
    (R)  (R)
  End Beta Spline
```

```
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R) (R)	**Default**	None
	(R) is the temperature		
	(R) is the thermal expansion coefficient, β		

Description

A monotonic spline interpolation of the thermal expansion coefficient, β, for the material as a function of temperature.

Example

10.2.25 `Beta Polynomial`

```
Begin Material (S)
  ...
  Beta Polynomial
    (R...)
    range = (R)  (R)
  End Beta Polynomial
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R...)	**Default**	None
	(R...) are the coefficients of the polynomial		
	(R) (R) is the optional range of the polynomial		

Description

A polynomial is defined for the thermal expansion coefficient, β, of the material as a function of temperature. The range can be specified, if desired.

Example

10.2.26 `Pr =`

```
Begin Material (S)
  ...
  Pr = (R)
  ...
End Material (S)
```

Type	Line Command		
Scope	`Material` block		
Parameters	(R)	**Default**	None
	(R) is the Prandtl number		

Description

A constant Prandtl number is specified.

Example

10.2.27 `Pr Table`

```
Begin Material (S)
  ...
  Pr Table
!    T   Pr
    (R)  (R)
  End Pr Table
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R) (R)	**Default**	None
	(R) is the temperature		
	(R) is the Prandtl number		

Description

A piecewise linear interpolation of the Prandtl number for the material as a function of temperature.

Example

10.2.28 `Pr Spline`

```
Begin Material (S)
  ...
  Pr Spline
!    T    k
    (R)  (R)
  End Pr Spline
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R) (R)	**Default**	None
	(R) is the temperature		
	(R) is the Prandtl number		

Description

A monotonic spline interpolation of the Prandtl number for the material as a function of temperature.

Example

10.2.29 `Pr Polynomial`

```
Begin Material (S)
  ...
  Pr Polynomial
    (R...)
    range = (R)  (R)
  End Pr Polynomial
  ...
End Material (S)
```

Type	Block Command		
Scope	`Material` block		
Parameters	(R...)	**Default**	None
	(R...) are the coefficients of the polynomial		
	(R) (R) is the optional range of the polynomial		

Description

A polynomial is defined for the Prandtl number of the material as a function of temperature. The range can be specified, if desired.

Example

10.2.30 Gas Constant

```
Begin Material (S)
  ...
  Gas Constant = (R)
  ...
End Material (S)
```

Type	Line Command		
Scope	`Material` block		
Parameters	(R)	**Default**	None
	(R) is the gas constant		

Description

The gas constant is specified for use with ideal gas properties.

Example

10.2.31 Reference

```
Begin Material (S)
  ...
  Reference = (S ...)
  ...
End Material (S)
```

Type	Line Command		
Scope	`Material` block		
Parameters	(S . . .)	**Default**	None
	(R) is the material properties reference		

Description

The reference for the material properties is provided.

Example

Chapter 11

Summary of Input Commands

This chapter provides a summary of all the input commands to TNSolver.

```
Begin Solution Parameters

  title                         = (S ...)
  type                          = {<steady>|transient}
  units                         = {<SI>|US}
  T units                       = {<C>|K|F|R}
  nonlinear convergence         = {<1.0E-9>|(R)}
  maximum nonlinear iterations = {<100>|(I)}
  begin time                    = {<0.0>|(R)}
  end time                      = (R)
  time step                     = (R)
  number of time steps          = (I)
  print interval                = {<1>|(I)}
  Stefan-Boltzmann              = {<5.6704E-8 W/m^2-K^4>|1.714e-9 Btu/hr-ft^2-R^4}
  gravity                       = {<9.80665 m/s^2>|32.174 ft/s^2}
  graphviz output               = {<no>|yes}

End Solution Parameters

Begin Nodes

! label  material  volume
   (S)     (S)      (R)

! label  density*specific heat  volume
   (S)          (R)              (R)

End Nodes

Begin Conductors
```

```
! label    type       nd_i  nd_j  parameters
   (S)  conduction    (S)   (S)  (R) (R) (R)       ! k, L, A
   (S)  conduction    (S)   (S)  (S) (R) (R)       ! material, L, A
   (S)  cylindrical   (S)   (S)  (R) (R) (R) (R)   ! k, ri, ro, L
   (S)  cylindrical   (S)   (S)  (S) (R) (R) (R)   ! material, ri, ro, L
   (S)  spherical     (S)   (S)  (R) (R) (R)       ! k, ri, ro
   (S)  spherical     (S)   (S)  (S) (R) (R)       ! material, ri, ro

   (S)  convection    (S)   (S)  (R) (R)           ! h, A
   (S)  IFCduct       (S)   (S)  (S) (R) (R) (R)   ! material, velocity, Dh, A
   (S)  EFCcyl        (S)   (S)  (S) (R) (R) (R)   ! material, velocity, D, A
   (S)  EFCdiamond    (S)   (S)  (S) (R) (R) (R)   ! material, velocity, D, A
   (S)  EFCimpjet     (S)   (S)  (S) (R) (R) (R) (R) ! material, velocity,
                                                     ! D, H, r
   (S)  EFCplate      (S)   (S)  (S) (R) (R) (R) (R) ! material, velocity,
                                                     ! Xbegin, Xend, A
   (S)  EFCsphere     (S)   (S)  (S) (R) (R)       ! material, velocity, D
   (S)  INCvenc       (S)   (S)  (S) (R) (R) (R)   ! material, W, H, A
   (S)  ENChcyl       (S)   (S)  (S) (R) (R)       ! material, D, A
   (S)  ENChplatedown (S)   (S)  (S) (R) (R)       ! material, L=A/P, A
   (S)  ENChplateup   (S)   (S)  (S) (R) (R)       ! material, L=A/P, A
   (S)  ENCiplatedown (S)   (S)  (S) (R) (R) (R) (R) ! material, H, L=A/P,
                                                     ! angle, A
   (S)  ENCiplateup   (S)   (S)  (S) (R) (R) (R) (R) ! material, H, L=A/P,
                                                     ! angle, A
   (S)  ENCsphere     (S)   (S)  (S) (R)           ! material, D
   (S)  ENCvplate     (S)   (S)  (S) (R) (R)       ! material, L, A
   (S)  FCuser        (S)   (S)  (S) (S) (R ...) (R) ! function, material,
                                                     ! parameters, A
   (S)  NCuser        (S)   (S)  (S) (S) (R ...) (R) ! function, material,
                                                     ! parameters, A

   (S)  surfrad       (S)   (S)  (R) (R)           ! emissivity, A
   (S)  radiation     (S)   (S)  (R) (R)           ! script-F, A

   (S)  advection     (S)   (S)  (S) (R) (R)       ! material, velocity, A

End Conductors

Begin Boundary Conditions

! type       parameter  node(s)
  fixed_T    (R)        (S ...)  ! T
  heat_flux  (R) (R)    (S ...)  ! q, A

End Boundary Conditions
```

```
Begin Sources

! type   parameter(s)    node(s)
  qdot   (R)             (S ...)  ! q̇, uses node volume: Q = q̇V
  Qsrc   (R)             (S ...)  ! Q
  tstatQ (R) (S) (R) (R) (S ...)  ! Q, thermostat node, Ton, Toff

End Sources

Begin Initial Conditions

! Initial T  node(s)
     (R)     all      ! apply to all nodes in the model
     (R)     (S ...)
  read restart = (S)  ! read T from restart file

End Initial Conditions

Begin Radiation Enclosure

! label emiss  area  view factors
   (S)   (R)   (R)   (R ...)

End Radiation Enclosure

Begin Functions

  Begin Constant (S)  ! function name
    (R)
  End Constant (S)

  Begin Time {Table|Spline} (S)  ! function name
!  time value
    (R)  (R)
    ...
    (R)  (R)
  End Time {Table|Spline} (S)

  Begin Polynomial (S)  ! function name
    (R ...)             ! a0 a1 a2 ...
    range = (R) (R)     ! range begin and end
  End Polynomial (S)

  Begin Composite (S)  ! function name
    (S ...)            ! list of function names
  End Composite (S)
```

```
End Functions

Begin Material (S)   ! material name

  State = (S)  ! {gas|liquid|solid}, required for all materials

  Density = {(R)|ideal gas}  ! ideal gas: ρ = P/RT
  Density {Table|Spline}
!    T  density
    (R)  (R)
    ...
    (R)  (R)
  End Density {Table|Spline}
  Density Polynomial
    (R ...)          ! a0 a1 a2 ...
    range = (R) (R)  ! range begin and end
  End Density Polynomial

  Conductivity = (R)
  Conductivity {Table|Spline}
!    T    k
    (R)  (R)
    ...
    (R)  (R)
  End Conductivity {Table|Spline}
  Conductivity Polynomial
    (R ...)          ! a0 a1 a2 ...
    range = (R) (R)  ! range begin and end
  End Conductivity Polynomial

  {Specific Heat|c_v} = (R)  ! constant volume specific heat
  {Specific Heat|c_v} {Table|Spline}
!    T   c_v
    (R)  (R)
    ...
    (R)  (R)
  End {Specific Heat|c_v} {Table|Spline}
  {Specific Heat|c_v} Polynomial
    (R ...)          ! a0 a1 a2 ...
    range = (R) (R)  ! range begin and end
  End {Specific heat|c_v} Polynomial

  c_p = (R)  ! constant pressure specific heat
  c_p {Table|Spline}
!    T   c_p
    (R)  (R)
```

```
    ...
    (R)  (R)
  End c_p {Table|Spline}
  c_p Polynomial
    (R ...)          ! a0 a1 a2 ...
    range = (R) (R)  ! range begin and end
  End c_p Polynomial

  Viscosity = (R)   ! dynamic viscosity: μ
  Viscosity {Table|Spline}
!    T  viscosity, μ
    (R)  (R)
    ...
    (R)  (R)
  End Viscosity {Table|Spline}
  Viscosity Polynomial
    (R ...)          ! a0 a1 a2 ...
    range = (R) (R)  ! range begin and end
  End Viscosity Polynomial

  Beta = {(R)|ideal gas}  ! thermal expansion coefficient,
                          ! ideal gas: β = 1/T
  Beta {Table|Spline}
!    T   beta, β
    (R)  (R)
    ...
    (R)  (R)
  End Beta {Table|Spline}
  Beta Polynomial
    (R ...)          ! a0 a1 a2 ...
    range = (R) (R)  ! range begin and end
  End Beta Polynomial

  Pr = (R)                ! Prandtl number, Pr = c_p μ/k
  Pr {Table|Spline}
!    T   Pr, Pr = c_p μ/k
    (R)  (R)
    ...
    (R)  (R)
  End Pr {Table|Spline}
  Pr Polynomial
    (R ...)          ! a0 a1 a2 ...
    range = (R) (R)  ! range begin and end
  End Pr Polynomial

  gas constant = (R)  ! gas constant for use with ideal gas: R = R̂/M
```

```
  reference = (S ...)

End Material (S)
```

Chapter 12

Output Files

Results output from TNSolver consist of three ASCII text files. The file with the extension .out contains a summary of the results. There are two comma-separated values (CSV) output files. One is for nodal data and the other is for conductor data. The CSV files can be imported by a spreadsheet program, such as Microsoft Excel, LibreOffice Calc or Apache OpenOffice Calc.

12.1 Results Output

An example output file is shown in Figure 12.1.

12.2 Node CSV Output

The node CSV output file contains the node data for the simulation results. An example is shown in Figure 12.2. The data in the file shows the node label, material name, volume and temperature. Note that a material name of `N/A` indicates that the thermal capacitance, ρc_v, of the node was supplied, instead of using a named material.

12.3 Conductor CSV Output

U is the *thermal conductance* or the *overall heat transfer coefficient*, defined in Table 12.1:

$$Q_{ij} = UA(T_i - T_j) \tag{12.1}$$

Table 12.1: Definitions of Thermal Conductance

Conductor	Thermal Conductance, U
conduction	k/L
convection	h
surfrad	$h_r = \epsilon\sigma(T_i + T_j)(T_i^2 + T_j^2)$
EFCcyl	h
EFCdiamond	h
ENChcyl	h

12.4 Transient Data CSV Output

For a transient simulation the nodal temperature and conductor heat flow rates are stored in the time data CSV file. An example is shown in Figure 12.4.

12.5 Restart File

At the end of each simulation TNSolver will write a restart file using the nodal temperatures. An example of a restart file is shown in Figure 12.5.

12.6 graphviz dot File

If requested in the `Solution Parameters` command block using the `graphviz output` line command a graphviz dot file will be written. The name of the file will be the base name of the input file with `.gv` as its extension. Using the graphviz software a visualization of the solution is produced as shown in Figure 12.6, using the dot command:

```
>dot -Tpdf -rt0001.gv -o rt0001.pdf
```

The input file (`rt0001.inp`) for this model is:

```
Begin Solution Parameters

  title = Regression Test 0001
  type = steady
  !type = transient
  begin time = 0.0
  end time = 0.01
  number of time steps = 10
  nonlinear convergence = 1.0e-8
  maximum nonlinear iterations = 15
  graphviz output = yes

End Solution Parameters

Begin Nodes

! label  material  volume
    1      N/A      0.0
    2      steel    1.74
    3      N/A      0.0
    4      air      2.58
    5      N/A      0.0
```

```
    6      fir      3.85
    7      N/A      0.0

End Nodes

Begin Conductors

! label  type        node 1  node 2  parameters
   10    conduction    1       2     steel  0.67  1.0  ! mat L A
   11    conduction    2       3     steel  0.67  1.0  ! mat L A
   12    convection    3       4     12.5  1.0         ! h A
   13    convection    4       5     23.8  1.0         ! h A
   14    radiation     3       5     0.19565 1.0       ! script-F A
   15    conduction    5       6     fir    1.29  1.0  ! mat L A
   16    conduction    6       7     fir    1.29  1.0  ! mat L A
   17    radiation     7     T_r     0.9    1.0        ! script-F A
   18    convection    7     T_c     2.3  1.0        ! h A
   19    convection    1     T_c     2.3  1.0        ! h A

End Conductors

Begin Boundary Conditions

! type    parameter(s)   node(s)
 fixed_T    -20.0         T_r    ! Far field radiation T
 fixed_T     20.0         T_c    ! Convection fluid T
 heat_flux    7.3  1.0    1      ! q A

End Boundary Conditions

Begin Sources

! type  parameter(s)  node(s)
  qdot    0.9          2         ! qdot

End Sources

Begin Initial Conditions

! Initial T   Node(s)
    23.5      all
    40.0      3  5

End Initial Conditions
```

```
**********************************************************
*                                                        *
*          TNSolver - A Thermal Network Solver           *
*                                                        *
*             Version 0.6.x, April 20, 2016              *
*                                                        *
**********************************************************

Model run finished at  2:18 PM, on July 17, 2016

*** Solution Parameters ***

  Title: Regression Test 0001

  Type                             =  steady
  Units                            =  SI
  Temperature units                =  C
  Nonlinear convergence            =  1e-08
  Maximum nonlinear iterations     =  15
  Gravity                          =  9.80665 (m/s^2)
  Stefan-Boltzmann constant        =  5.67037e-08 (W/m^2-K^4)

*** Nodes ***

                         Volume   Temperature
   Label    Material     (m^3)        (C)
 --------- ---------- ---------- -----------
         1        N/A          0        23.3
         2      steel       1.74     23.3031
         3        N/A          0     23.2897
         4        air       2.58     23.2003
         5        N/A          0     23.1533
         6        fir       3.85     9.43722
         7        N/A          0    -4.27883
       T_c        N/A          0          20
       T_r        N/A          0         -20

*** Conductors ***

                                                    Q_ij
    Label        Type       Node i     Node j       (W)
 ---------- ------------- ---------- ---------- ----------
         10    conduction          1          2  -0.290088
         11    conduction          2          3    1.27591
         12    convection          3          4    1.11828
         13    convection          4          5    1.11828
         14     radiation          3          5   0.157628
         15    conduction          5          6    1.27591
         16    conduction          6          7    1.27591
         17     radiation          7        T_r    57.1172
         18    convection          7        T_c   -55.8413
         19    convection          1        T_c    7.59009
```

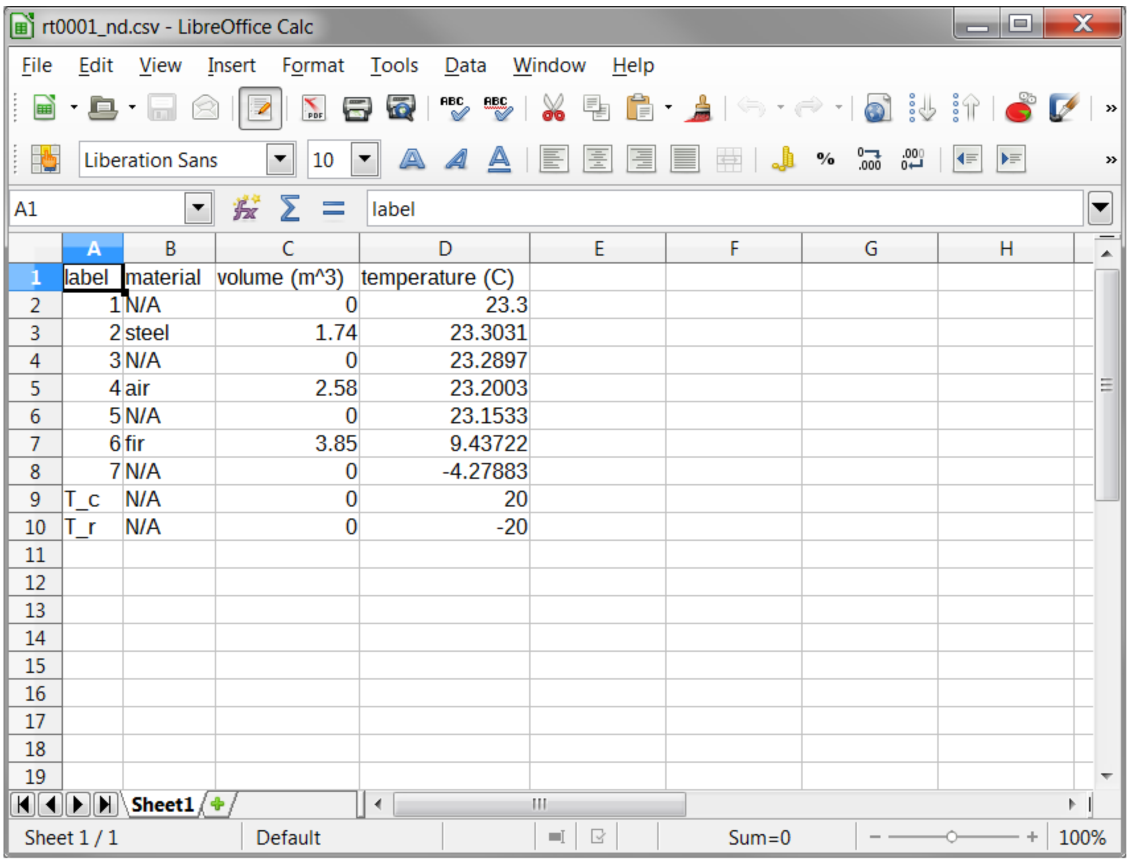
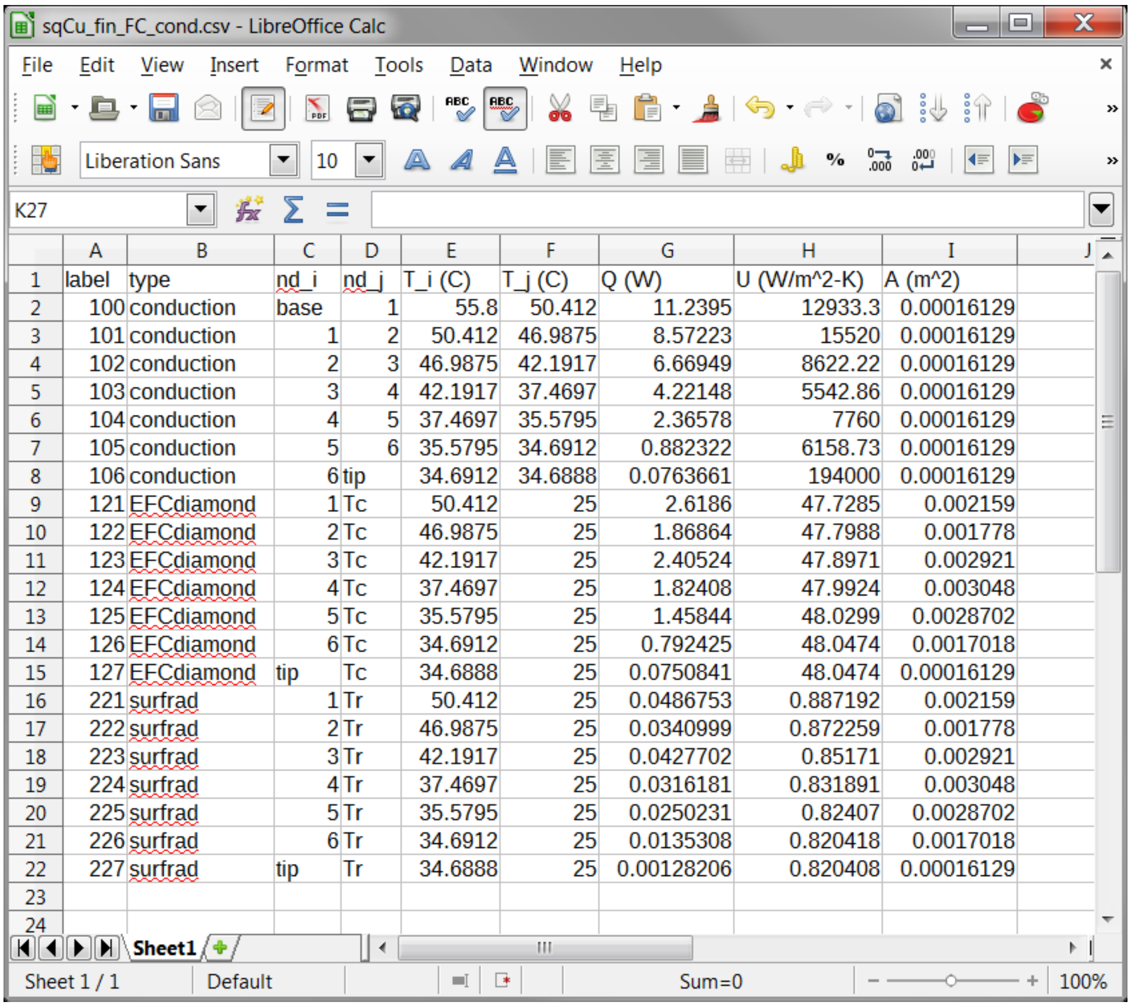
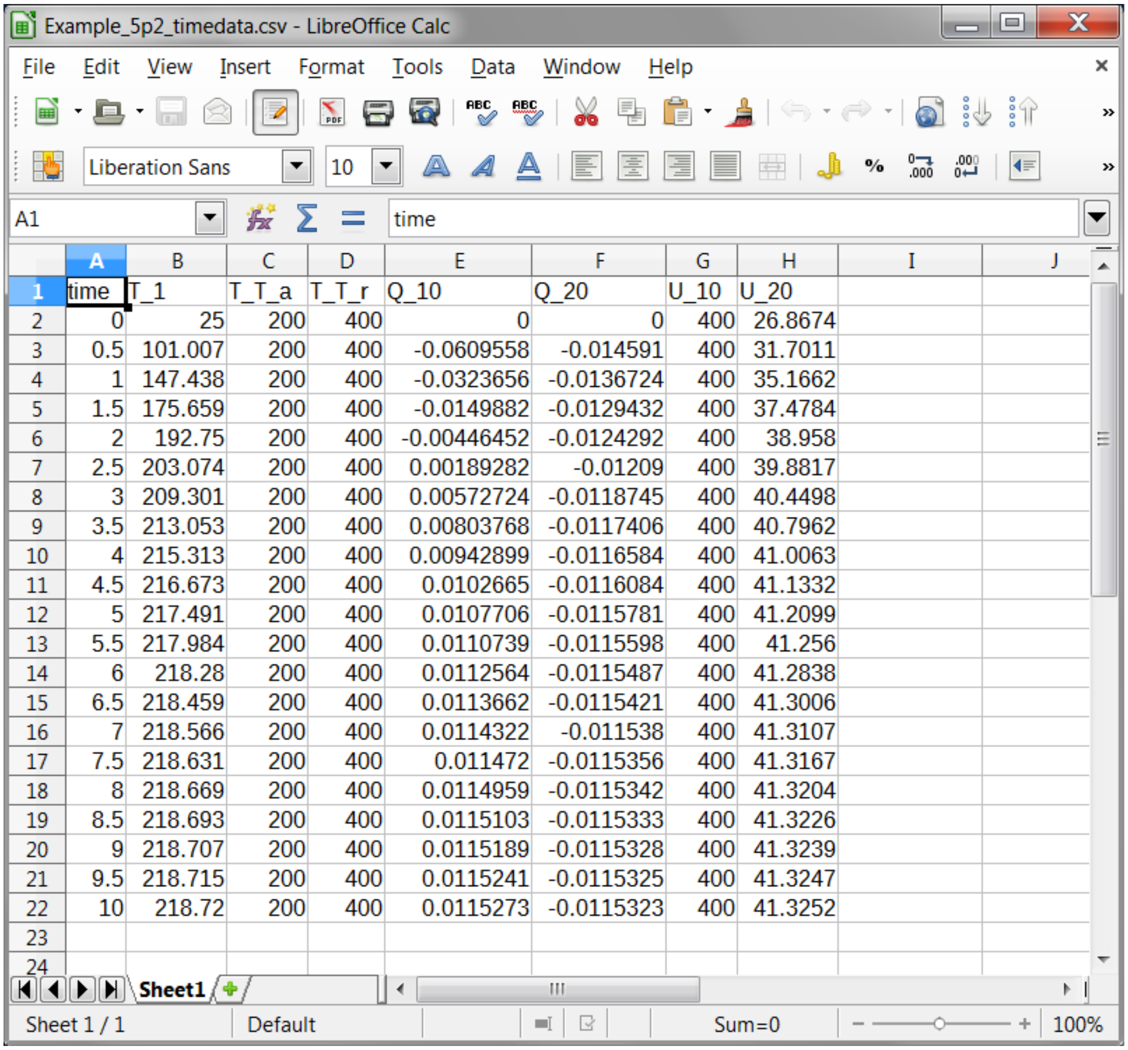


Figure 12.2: Node CSV File Imported Into LibreOffice Calc



label	type	nd_i	nd_j	T_i (C)	T_j (C)	Q (W)	U (W/m^2-K)	A (m^2)
100	conduction	base	1	55.8	50.412	11.2395	12933.3	0.00016129
101	conduction	1	2	50.412	46.9875	8.57223	15520	0.00016129
102	conduction	2	3	46.9875	42.1917	6.66949	8622.22	0.00016129
103	conduction	3	4	42.1917	37.4697	4.22148	5542.86	0.00016129
104	conduction	4	5	37.4697	35.5795	2.36578	7760	0.00016129
105	conduction	5	6	35.5795	34.6912	0.882322	6158.73	0.00016129
106	conduction	6	tip	34.6912	34.6888	0.0763661	194000	0.00016129
121	EFCdiamond	1	Tc	50.412	25	2.6186	47.7285	0.002159
122	EFCdiamond	2	Tc	46.9875	25	1.86864	47.7988	0.001778
123	EFCdiamond	3	Tc	42.1917	25	2.40524	47.8971	0.002921
124	EFCdiamond	4	Tc	37.4697	25	1.82408	47.9924	0.003048
125	EFCdiamond	5	Tc	35.5795	25	1.45844	48.0299	0.0028702
126	EFCdiamond	6	Tc	34.6912	25	0.792425	48.0474	0.0017018
127	EFCdiamond	tip	Tc	34.6888	25	0.0750841	48.0474	0.00016129
221	surfrad	1	Tr	50.412	25	0.0486753	0.887192	0.002159
222	surfrad	2	Tr	46.9875	25	0.0340999	0.872259	0.001778
223	surfrad	3	Tr	42.1917	25	0.0427702	0.85171	0.002921
224	surfrad	4	Tr	37.4697	25	0.0316181	0.831891	0.003048
225	surfrad	5	Tr	35.5795	25	0.0250231	0.82407	0.0028702
226	surfrad	6	Tr	34.6912	25	0.0135308	0.820418	0.0017018
227	surfrad	tip	Tr	34.6888	25	0.00128206	0.820408	0.00016129

Figure 12.3: Conductor CSV File Imported Into LibreOffice Calc



time	T_1	T_T_a	T_T_r	Q_10	Q_20	U_10	U_20
0	25	200	400	0	0	400	26.8674
0.5	101.007	200	400	-0.0609558	-0.014591	400	31.7011
1	147.438	200	400	-0.0323656	-0.0136724	400	35.1662
1.5	175.659	200	400	-0.0149882	-0.0129432	400	37.4784
2	192.75	200	400	-0.00446452	-0.0124292	400	38.958
2.5	203.074	200	400	0.00189282	-0.01209	400	39.8817
3	209.301	200	400	0.00572724	-0.0118745	400	40.4498
3.5	213.053	200	400	0.00803768	-0.0117406	400	40.7962
4	215.313	200	400	0.00942899	-0.0116584	400	41.0063
4.5	216.673	200	400	0.0102665	-0.0116084	400	41.1332
5	217.491	200	400	0.0107706	-0.0115781	400	41.2099
5.5	217.984	200	400	0.0110739	-0.0115598	400	41.256
6	218.28	200	400	0.0112564	-0.0115487	400	41.2838
6.5	218.459	200	400	0.0113662	-0.0115421	400	41.3006
7	218.566	200	400	0.0114322	-0.011538	400	41.3107
7.5	218.631	200	400	0.011472	-0.0115356	400	41.3167
8	218.669	200	400	0.0114959	-0.0115342	400	41.3204
8.5	218.693	200	400	0.0115103	-0.0115333	400	41.3226
9	218.707	200	400	0.0115189	-0.0115328	400	41.3239
9.5	218.715	200	400	0.0115241	-0.0115325	400	41.3247
10	218.72	200	400	0.0115273	-0.0115323	400	41.3252

Figure 12.4: Time Data CSV File Imported Into LibreOffice Calc

```
time = 0
! node  T
    1  23.3
    2  23.3031
    3  23.2897
    4  23.2003
    5  23.1533
    6  9.43722
    7  -4.27883
  T_c  20
  T_r  -20
```

Figure 12.5: Example Restart File

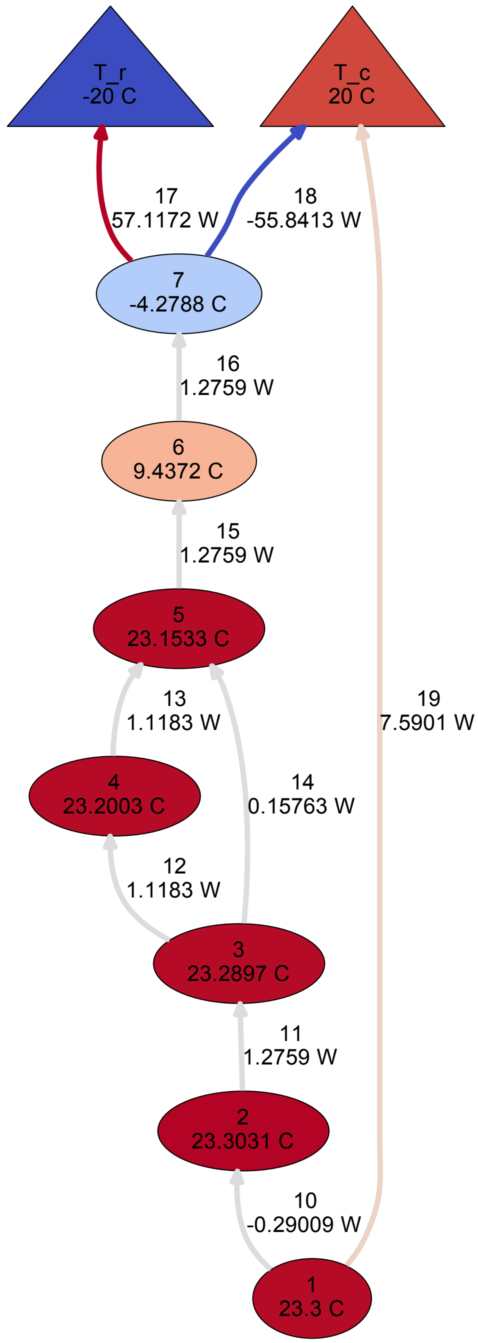


Figure 12.6: Solution Visualization using graphviz

Chapter 13

Examples

13.1 Simple Plane Wall

This example model is of steady heat transfer through a plane wall with a specified temperature on the inner wall and convection to a fluid on the outer wall. The thermal network is shown in Figure 13.1. The input file is shown in Figure 13.2. A title is supplied and the simulation type is set to steady. There are three nodes in the model, with the labels `in` `out` and `Tinf`. A conduction conductor labeled `wall`, with a specified thermal conductivity of $k = 2.3\ W/m \cdot K$, a wall thickness of 0.1524 m. The total heat flow rate per square meter is desired, so the area is $A = 1.0\ m^2$.

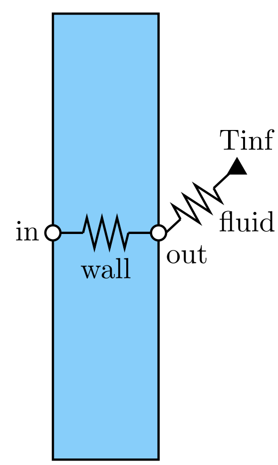


Figure 13.1: Plane Wall

13.2 Composite Wall

Consider a composite wall as shown in Figure 13.3.

```
!  Simple Wall Model

Begin Solution Parameters

  title = Simple Wall Model
  type = steady

End Solution Parameters

Begin Conductors

! label type       nd_i nd_j  parameters
  wall  conduction  in  out   2.3  1.2   1.0 ! k L A
  fluid convection  out Tinf  2.3  1.0       ! h A

End Conductors

Begin Boundary Conditions

!  type   parameter  node(s)
  fixed_T  21.0      in    ! Inner wall T
  fixed_T  5.0       Tinf  ! Fluid T

End Boundary Conditions
```

Figure 13.2: Simple Wall Model Input File

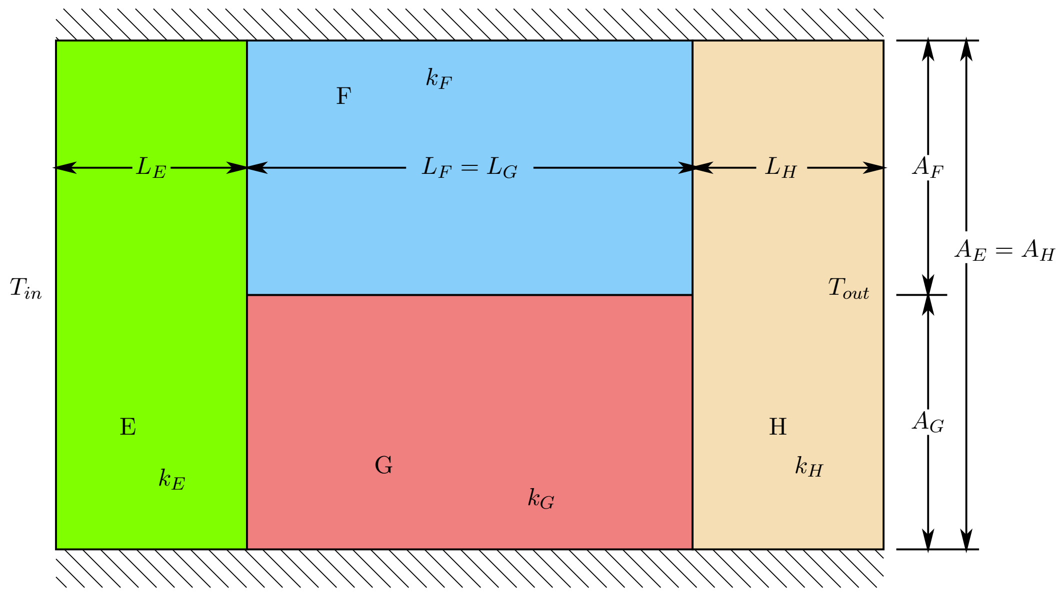


Figure 13.3: Composite Wall

Appendix A

Units

The following sections present nomenclature and sets of consistent units based on the fundamental units of mass (M), length (L), time (t), temperature (T) and amount of substance (N), see [Car03] and [TT08].

A.1 International System of Units (SI)

The consistent set of units for the international system (SI) are given in Table A.1. Unit prefixes commonly used in the SI system of units are shown in Table A.2.

A.2 English System of Units

The consistent set of units for the English system of units (US Customary units) are given in Table A.3.

A.3 Conversion of Units

The following tables provide conversion factors between the two standard unit systems in TNSolver.

Table A.1: SI System of Units

Quantity	Symbol		Fundamental	Derivatives
Mass	m	M	kg	
Length	x, y, z	L	m	
Area	A	L^2	m^2	
Volume	V	L^3	m^3	
Time	t	t	s	
Force	F	$\frac{M \cdot L}{t^2}$	$\frac{kg \cdot m}{s^2}$	newton (N)
Energy	E	$\frac{M \cdot L^2}{t^2}$	$\frac{kg \cdot m^2}{s^2}$	joule (J), $N \cdot m$
Power	P	$\frac{M \cdot L^2}{t^3}$	$\frac{kg \cdot m^2}{s^3}$	watt (W), $\frac{J}{s}$
Rate of heat transfer	$Q = qA$	$\frac{M \cdot L^2}{t^3}$	$\frac{kg \cdot m^2}{s^3}$	watt (W), $\frac{J}{s}$
Heat flux	q	$\frac{M}{t^3}$	$\frac{kg}{s^3}$	$\frac{W}{m^2}, \frac{J}{s \cdot m^2}$
Heat generation rate per unit volume	$\dot{q}$	$\frac{M}{L \cdot t^3}$	$\frac{kg}{m \cdot s^3}$	$\frac{W}{m^3}$, $\frac{J}{s \cdot m^3}$
Temperature	T	T	K	$°C = K - 273.15$
Pressure	P	$\frac{M}{L \cdot t^2}$	$\frac{kg}{m \cdot s^2}$	pascal (Pa), $\frac{N}{m^2}$
Velocity	u, v, w	$\frac{L}{t}$	$\frac{m}{s}$	
Density	ρ	$\frac{M}{L^3}$	$\frac{kg}{m^3}$	
Thermal conductivity	k	$\frac{M \cdot L}{t^3 \cdot T}$	$\frac{kg \cdot m}{s^3 \cdot K}$	$\frac{W}{m \cdot K}$
Specific heat	c	$\frac{L^2}{t^2 \cdot T}$	$\frac{m^2}{s^2 \cdot K}$	$\frac{J}{kg \cdot K}$
Dynamic (absolute) viscosity	μ	$\frac{M}{L \cdot t}$	$\frac{kg}{m \cdot s}$	$Pa \cdot s$, $\frac{N \cdot s}{m^2}$
Volumetric thermal expansion coefficient	β	$\frac{1}{T}$	$\frac{1}{K}$	
Thermal diffusivity	$\alpha = \frac{k}{\rho c}$	$\frac{L^2}{t}$	$\frac{m^2}{s}$	
Kinematic Viscosity	$\nu = \frac{\mu}{\rho}$	$\frac{L^2}{t}$	$\frac{m^2}{s}$	
Convective heat transfer coefficient	h	$\frac{M}{t^3 \cdot T}$	$\frac{kg}{s^3 \cdot K}$	$\frac{W}{m^2 \cdot K}$, $\frac{J}{s \cdot m^2 \cdot K}$

Table A.2: SI Unit Prefixes

Prefix	Symbol	Factor
tera	T	$\times 1,000,000,000,000$ (10^{12})
giga	G	$\times 1,000,000,000$ (10^9)
mega	M	$\times 1,000,000$ (10^6)
kilo	k	$\times 1,000$ (10^3)
deci	d	$\times 1/10$ (10^{-1})
centi	c	$\times 1/100$ (10^{-2})
milli	m	$\times 1/1,000$ (10^{-3})
micro	μ	$\times 1/1,000,000$ (10^{-6})
nano	n	$\times 1/1,000,000,000$ (10^{-9})
pico	p	$\times 1/1,000,000,000,000$ (10^{-12})

Table A.3: English System of Units

Quantity	Symbol		Fundamental	Derivatives
Mass	m	M	lb_m	
Length	x, y, z	L	ft	
Area	A	L^2	ft^2	
Volume	V	L^3	ft^3	
Time	t	t	s	
Force	F	$\frac{M \cdot L}{t^2}$	$\frac{lb_m \cdot ft}{s^2}$	$lb_f = 32.174 \frac{lb_m \cdot ft}{s^2}$
Energy	E	$\frac{M \cdot L^2}{t^2}$	$\frac{lb_m \cdot ft^2}{s^2}$	Btu
Power	P	$\frac{M \cdot L^2}{t^3}$	$\frac{lb_m \cdot ft^2}{s^3}$	hp
Rate of heat transfer	$Q = qA$	$\frac{M \cdot L^2}{t^3}$	$\frac{lb_m \cdot ft^2}{s^3}$	$\frac{Btu}{s}$
Heat flux	q	$\frac{M}{t^3}$	$\frac{lb_m}{s^3}$	$\frac{Btu}{s \cdot ft^2}$
Heat generation rate per unit volume	$\dot{q}$	$\frac{M}{L \cdot t^3}$	$\frac{lb_m}{ft \cdot s^3}$	$\frac{Btu}{s \cdot ft^3}$
Temperature	T	T	$^\circ R$	$^\circ F = {^\circ R} - 459.67$
Pressure	P	$\frac{M}{L \cdot t^2}$	$\frac{lb_m}{ft \cdot s^2}$	
Velocity	u, v, w	$\frac{L}{t}$	$\frac{ft}{s}$	
Density	ρ	$\frac{M}{L^3}$	$\frac{lb_m}{ft^3}$	
Thermal conductivity	k	$\frac{M \cdot L}{t^3 \cdot T}$	$\frac{lb_m \cdot ft}{s^3 \cdot {^\circ R}}$	$\frac{Btu}{s \cdot ft \cdot {^\circ R}}$
Specific heat	c	$\frac{L^2}{t^2 \cdot T}$	$\frac{ft^2}{s^2 \cdot {^\circ R}}$	$\frac{Btu}{lb_m \cdot {^\circ R}}$
Dynamic (absolute) viscosity	μ	$\frac{M}{L \cdot t}$	$\frac{lb_m}{ft \cdot s}$	
Volumetric thermal expansion coefficient	β	$\frac{1}{T}$	$\frac{1}{^\circ R}$	
Thermal diffusivity	$\alpha = \frac{k}{\rho c}$	$\frac{L^2}{t}$	$\frac{ft^2}{s}$	
Kinematic Viscosity	$\nu = \frac{\mu}{\rho}$	$\frac{L^2}{t}$	$\frac{ft^2}{s}$	
Convective heat transfer coefficient	h	$\frac{M}{t^3 \cdot T}$	$\frac{lb_m}{s^3 \cdot {^\circ R}}$	$\frac{Btu}{s \cdot ft^2 \cdot {^\circ R}}$

Table A.4: Practical Values

Temperature		Velocity		Pressure	
F	C	mph	m/s	psi	Pa
0.0	-17.8	1.0	0.44704	1.0	6,894.8
32.0	0.0	5.0	2.2352	5.0	34,473.8
70.0	21.1	10.0	4.4704	14.696	101,325.0
100.0	37.8	20.0	8.9408	50.0	344,737.9
212.0	100.0	50.0	22.352	100.0	689,475.7
		100.0	44.704		

Table A.5: SI to US Units Conversion Factors [BCG06]

Quantity	SI	Multiply by	US
time, t	s	$\times 0.000277778 =$	hr
length, L	m	$\times 3.2808399 =$	ft
area, A	m^2	$\times 10.76391 =$	ft^2
volume, V	m^3	$\times 35.314667 =$	ft^3
temperature, T	K	$\times 1.8 =$	$^\circ R$
density, ρ	$\frac{kg}{m^3}$	$\times 0.062427961 =$	$\frac{lb_m}{ft^3}$
thermal conductivity, k	$\frac{W}{m \cdot K}$	$\times 0.57778932 =$	$\frac{Btu}{hr \cdot ft \cdot ^\circ R}$
specific heat, c_v, c_p	$\frac{J}{kg \cdot K}$	$\times 0.0002388459 =$	$\frac{Btu}{lb_m \cdot ^\circ R}$
viscosity, μ	$Pa \cdot s$ or $\frac{N \cdot s}{m^2}$	$\times 2419.0883 =$	$\frac{lb_m}{ft \cdot hr}$
thermal expansion, β	$\frac{1}{K}$	$\times 0.555556 =$	$\frac{1}{^\circ R}$
convection coefficient, h	$\frac{W}{m^2 \cdot K}$ or $\frac{J}{s \cdot m^2 \cdot K}$	$\times 0.17611018 =$	$\frac{Btu}{hr \cdot ft^2 \cdot ^\circ R}$
heat flux, q	$\frac{W}{m^2}$ or $\frac{J}{s \cdot m^2}$	$\times 0.31699833 =$	$\frac{Btu}{hr \cdot ft^2}$
rate of heat transfer, $Q = qA$	watt (W) or $\frac{J}{s}$	$\times 3.4121416 =$	$\frac{Btu}{hr}$

References

[BCG06] Kenneth Butcher, Linda Crown, and Elizabeth J. Gentry. The international system of units (si) – conversion factors for general use. Technical Report NIST Special Publication 1038, National Institute of Standards and Technology, Gaithersburg,MD, 2006.

[BLID11] T.L. Bergman, A.S. Lavine, F.P. Incropera, and D.P. DeWitt. *Introduction to Heat Transfer*. John Wiley & Sons, New York, sixth edition, 2011.

[Car03] François Cardarelli. *Encyclopaedia of Scientific Units, Weights and Measures: Their SI Equivalences and Origins*. Springer-Verlag, London, 2003.

[Cat78] I. Catton. Natural convection in enclosures. In *Heat Transfer 1978: Volume 6 Keynote Papers*, pages 13–31, Washington, 1978. National Research Council of Canada, Hemisphere Publishing Corporation.

[CC75] Stuart W. Churchill and Humbert H.S. Chu. Correlating equations for laminar and turbulent free convection from a vertical plate. *International Journal of Heat and Mass Transfer*, 18(11):1323–1329, 1975.

[EBHW15] John W. Eaton, David Bateman, Søren Hauberg, and Rik Wehbring. *GNU Octave version 4.0.0 manual: a high-level interactive language for numeric al computations*. 2015.

[Geb71] B. Gebhart. *Heat Transfer*. McGraw-Hill, New York, second edition, 1971.

[Gni13] V. Gnielinski. On heat transfer in tubes. *International Journal of Heat and Mass Transfer*, 63:134–140, 2013.

[Hot54] H. C. Hottel. Radiant-heat transmission. In *Heat Transmission* [McA54], chapter 4, pages 55–125.

[HS67] H. C. Hottel and A. F. Sarofim. *Radiative Transfer*. McGraw-Hill, New York, 1967.

[HSM11] J. R. Howell, R. Siegel, and M. P. Mengüç. *Thermal Radiation Heat Transfer*. CRC Press, Boca Raton, fifth edition, 2011.

[IB63] T. Ishimoto and J. T. Bevans. Method of evaluating script F for radiant exchange within an enclosure. *AIAA Journal*, 1(6):1428–1429, 1963.

[KK58] J. G. Knudsen and D. L. Katz. *Fluid Dynamics and Heat Transfer*. McGraw-Hill, New York, 1958.

[LL12] J. H. Lienhard, IV and J. H. Lienhard, V. *A Heat Transfer Textbook*. Phlogiston Press, Cambridge, Massachusetts, fourth edition, 2012. Available at: `http://ahtt.mit.edu`.

[Mar77] Holger Martin. Heat and mass transfer between impinging gas jets and solid surfaces. In James P. Hartnett and Jr. Thomas F. Irvine, editors, *Advances in Heat Transfer*, volume 13, pages 1–60. Academic Press, New York, 1977.

[McA54] W. H. McAdams. *Heat Transmission*. McGraw-Hill, New York, third edition, 1954.

[ME69] R. K. MacGregor and A. F. Emery. Free convection through vertical plane layers–moderate and high prandtl number fluids. *Journal of Heat Transfer*, 91(3):391–401, 1969.

[RH98] G. D. Raithby and K. G. T. Hollands. Natural convection. In Warren M. Rohsenow, James R Hartnett, and Young I. Cho, editors, *Handbook of Heat Transfer*, chapter four, pages 4.1–4.99. McGraw-Hill, New York, third edition, 1998.

[TT08] Barry N. Taylor and Ambler Thompson. The international system of units (SI). Technical Report NIST Special Publication 330, National Institute of Standards and Technology, Gaithersburg, MD, 2008.

[Whi72] Stephen Whitaker. Forced convection heat transfer correlations for flow in pipes, past flat plates, single cylinders, single spheres, and for flow in packed beds and tube bundles. *AIChE Journal*, 18(2):361–371, 1972.

Index