Introduction to Heat Transfer Analysis

Thermal Network Solutions with TNSolver

Bob Cochran
Applied Computational Heat Transfer
Seattle, WA
TNSolver@heattransfer.org

ME 331 Introduction to Heat Transfer
University of Washington
October 3, 2017

Outline

- Heat Transfer Analysis
- Introduction to TNSolver
- Steady Conduction Example
- Convection and Surface Radiation Example

Heat Transfer Methods

Heat Transfer Analysis

Conduction, Convection and Radiation

Help! The barn is on fire.

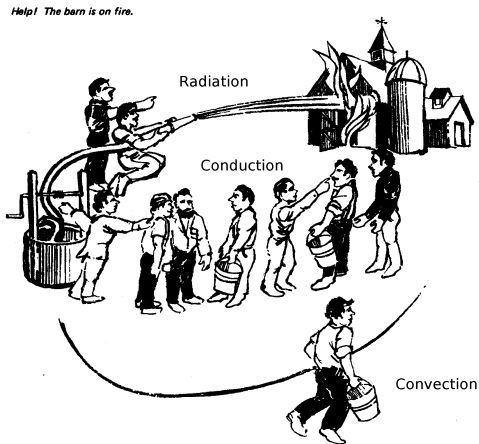Radiation

Conduction

Convection

Figure borrowed from [LL16].

Analysis Methods Summary

Heat Transfer Analysis

Answering design questions about thermal energy and temperature

- ▶ Hand calculation - back-of-the-envelope
  - ▶ On the order of 1-10 equations
- ▶ Spreadsheet style
  - ▶ Interactive Heat Transfer (IHT 4.0), see p. ix in [BLID11]
  - ▶ LibreOffice Calc, Microsoft Excel, MathCAD
- ▶ Thermal network or lumped parameter approach
  - ▶ On the order of 10-1,000 equations
- ▶ Continuum approach - solid model/mesh generation
  - ▶ On the order of 1,000-1,000,000 equations
  - ▶ Finite Volume Method (FVM)
  - ▶ Finite Element Method (FEM)

See Section 1.5, page 38, in [BLID11]

Overview of Analysis

Heat Transfer Analysis

- ▶ Energy conservation: control volumes
- ▶ Identify and sketch out the control volumes
- ▶ Use the conductor analogy to represent energy transfer between the control volumes and energy generation or storage
    - ▶ Conduction, convection, radiation, other?
    - ▶ Capacitance
    - ▶ Sources or sinks
- ▶ State assumptions and determine appropriate parameters for each conductor
    - ▶ Geometry, material properties, etc.
- ▶ Which conductor(s)/source(s)/capacitance(s) are important to the required results?
    - ▶ Sensitivity analysis
- ▶ What is missing from the model? - peer/expert review

Commercial Thermal Network Solvers

Heat Transfer Analysis

- ▶ C&R Technologies
  - ▶ SINDA/FLUINT, Thermal Desktop, RadCAD
- ▶ MSC Software
  - ▶ Sinda, SindaRad, Patran
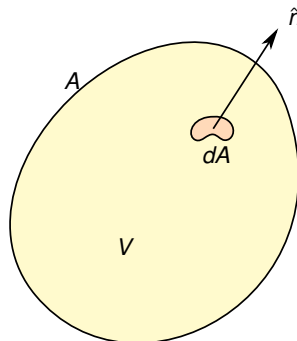- ▶ ESATAN-TMS
  - ▶ Thermal, Radiative, CADbench

The Control Volume Concept

Heat Transfer Analysis

$$\sum \text{Energy In} - \sum \text{Energy Out} =$$

Energy Stored, Generated and/or Consumed

Heat (transfer) is thermal energy transfer due to a temperature difference

Integral Form of Steady Heat Conduction
Heat Transfer Analysis

The steady conduction equation, in Cartesian tensor integral form, is:

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

where $\dot{q}$ is a volumetric source and Fourier's Law of Heat Conduction provides a constitutive model for the heat flux as a function of temperature gradient:

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

where k is the isotropic thermal conductivity.

Convection
Heat Transfer Analysis

Convection heat transfer from the surface of the control volume is modeled by:

$$\int_{\Gamma_c} q_i n_i \, dA = \int_{\Gamma_c} h(T_s - T_c) \, dA, \ \ \text{where} \ \left\{ \begin{array}{ll} T_s > T_c, & \text{cooling} \\ T_s < T_c, & \text{heating} \end{array} \right.$$

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

Surface Radiation
Heat Transfer Analysis

Radiation exchange between a surface and *large* surroundings
The heat flow rate is (Equation (1.7), page 10 in [BLID11]):

$$Q = \sigma \epsilon_s A_s (T_s^4 - T_{sur}^4)$$

where σ is the Stefan-Boltzmann constant, ϵ_s is the surface emissivity and A_s is the area of the surface.
Note that the surface area, A_s, must be *much* smaller than the surrounding surface area, A_{sur}:

$$A_s \ll A_{sur}$$

Note that the temperatures must be the absolute temperature, K or $^\circ R$

Radiation Heat Transfer Coefficient
Math Model

Define the radiation heat transfer coefficient, h_r (see Equation (1.9), page 10 in [BLID11]):

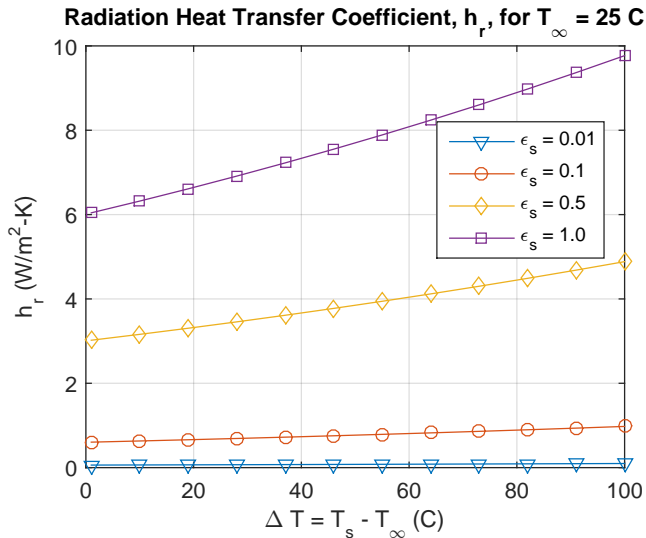$$h_r = \epsilon\sigma(T_s + T_{sur})(T_s^2 + T_{sur}^2)$$

Then,

$$Q = h_r A_s(T_s - T_{sur})$$

Note:

- h_r is temperature dependent
- h_r can be used to compare the radiation to the convection heat transfer from a surface, h (if T_{sur} and T_∞ have similar values)

Range of Radiation Heat Transfer Coefficient

Math Model

Radiation Heat Transfer Coefficient, h_r, for $T_\infty = 25$ C

- ► Thermal Network Solver - TNSolver
  - ► An open source implementation: you have complete access to what is happening behind the curtains
- ► MATLAB/Octave program
  - ► GNU Octave is an open source implementation of the MATLAB programming language
- ► Thermal model is described in a text input file
  - ► Do not use a word processor, use a text editor, such as:
    - ► Cross-platform: vim/gvim, emacs, Bluefish, among many others
    - ► Windows: notepad, Notepad++
    - ► MacOS: TextEdit, Smultron
    - ► Linux: see cross-platform options
- ► Simulation results are both returned from the function and written to text output files for post-processing

Example of Text Input File

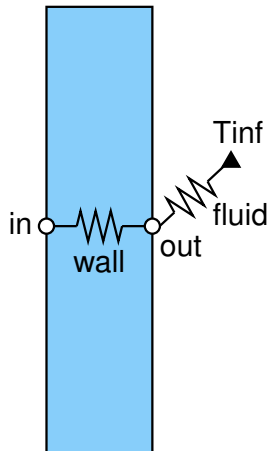TNSolver User Guide

```
!  Simple Wall Model

Begin Solution Parameters
type  = steady
End Solution Parameters

Begin Conductors
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
fixed_T  21.0   in    ! Inner wall T
fixed_T  5.0    Tinf  ! Fluid T
End Boundary Conditions
```


! begins a comment (MATLAB uses %)

Thermal Network Terminology

- ▶ Time dependency
    - ▶ Steady state or transient
    - ▶ Initial condition is required for transient
- ▶ Geometry
    - ▶ Control Volume - volume, $V = \int_V dV$
        - ▶ Node: ●, $T_{\text{node}} = \int_V T(x_i)dV$, finite volume
    - ▶ Control Volume Surface - area, $A = \int_A dA$
        - ▶ Surface Node: ○, $T_{\text{surface node}} = \int_A T(x_i)dA$, zero volume
- ▶ Material properties
- ▶ Conductors
    - ▶ Conduction
    - ▶ Convection
    - ▶ Radiation
- ▶ Boundary conditions
    - ▶ Boundary node: ▲
- ▶ Sources/sinks

Conduction: Cartesian (The Plane Wall)

The rate of heat transfer, Q_{ij}, due to conduction, between the two temperatures T_i and T_j, separated by a distance L and area A, is:

$$Q_{ij} = \frac{kA}{L}\left(T_i - T_j\right)$$

The heat flux, q_{ij}, is:

$$q_{ij} = \frac{Q_{ij}}{A} = \frac{k}{L}\left(T_i - T_j\right)$$

```
Begin Conductors

! label    type      node i node j  parameters
name conduction    label   label   x.x x.x x.x  ! k L A

End Conductors
```

Convection Conductor

TNSolver User Guide

The rate of heat transfer due to convection is:

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

```
Begin Conductors

! label   type       node i node j  parameters
name convection  label   label   x.x x.x   ! h A

End Conductors
```

Specified Surface Temperature Boundary Condition
TNSolver User Guide

The node temperature, T_b, is specified:

```
Begin Boundary Conditions

! type     parameter(s)  node(s)
fixed_T  T_b            label

End Boundary Conditions
```

Example 1.1, p. 5 in [BLID11]

Steady Conduction

Schematic:
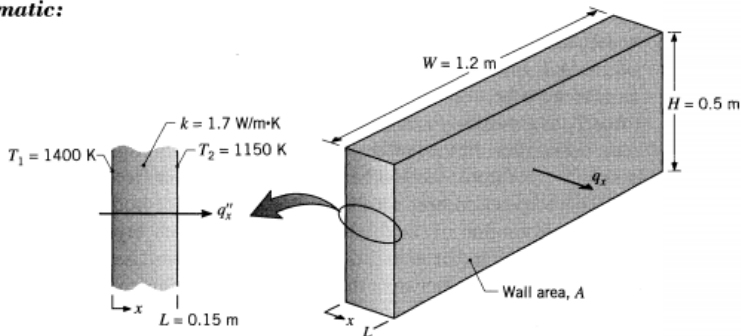
Figure borrowed from [BLID11].

TNSolver Input File

Steady Conduction

```
Begin Solution Parameters
  title = Example 1.1, p.5 in [BLID11]
  type  = steady
  units = SI
End Solution Parameters

Begin Conductors
! label type        nd_i nd_j  parameters
  wall  conduction  in   out  1.7  0.15 0.6 ! k, L, A
End Conductors

Begin Boundary Conditions
! type     parameter node
  fixed_T  1126.85   in   ! 1400 K (C = K - 273.15)
  fixed_T  876.85    out  ! 1150 K (C = K - 273.15)
End Boundary Conditions
```

Executing a TNSolver Model
Steady Conduction Example

The input file name is: `ex_1p1.inp`

```
>> [T, Q, nd, el] = tnsolver('ex_1p1');
```

T is a vector of node temperatures
Q is a vector of conductor heat flow rates
nd is a structure of node parameters
el is a structure of conductor parameters

TNSolver Output File

Steady Conduction Example

```
***********************************************************
*                                                         *
*         TNSolver - A Thermal Network Solver             *
*                                                         *
*             Version 0.9.2, August 9, 2017               *
*                                                         *
***********************************************************

Model run finished at 11:05 AM, on October 02, 2017

*** Solution Parameters ***

Title: Example 1.1, p.5 in [BLID11]

Type                          = steady
Units                         = SI
Temperature units             = C
Nonlinear convergence         = 1e-009
Maximum nonlinear iterations  = 100
Gravity                       = 9.80665 (m/s^2)
Stefan-Boltzmann constant     = 5.67037e-008 (W/m^2-K^4)
```

TNSolver Output File (continued)

Steady Conduction Example

```
*** Nodes ***

                    Volume    Temperature
 Label    Material   (m^3)       (C)
--------- ---------- ---------- -----------
      in       N/A          0     1126.85
     out       N/A          0      876.85

*** Conductors ***

                                                    Q_ij
   Label       Type        Node i     Node j       (W)
---------- -------------- ---------- ---------- ----------
     wall    conduction         in        out       1700

*** Boundary Conditions ***

   Type      Parameter(s)        Node(s)
---------- ------------------ --------------------
  fixed_T   1126.85            in
  fixed_T   876.85             out
```

Example 1.2, p. 10 in [BLID11]
Convection and Surface Radiation Example

Schematic:
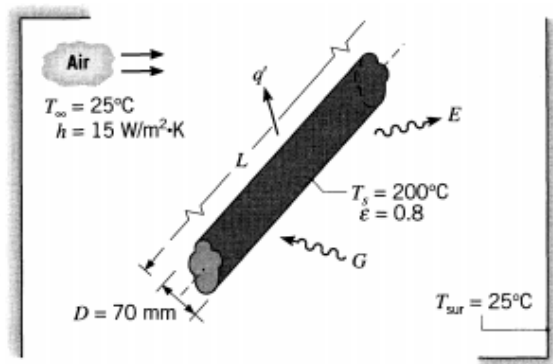

Figure borrowed from [BLID11].

Model Parameters

Convection and Surface Radiation Example

Pipe diameter: $D = 70mm = 0.07m$
Pipe surface area: $A = \pi DL = 3.14 * 0.07 * 1.0 = 0.22m^2$
Convection coefficient: $h = 15W/m^2 \cdot K$
Surface emissivity: $\epsilon = 0.8$

TNSolver Input File

Convection and Surface Radiation Example

```
Begin Solution Parameters
  title = Example 1.2, p. 10 in [BLID11]
  type  = steady
End Solution Parameters

Begin Conductors
! label type          nd_i nd_j  parameters
  conv   convection   surf wall  15.0 0.22  ! h, A
  rad    surfrad      surf wall   0.8 0.22  ! emissivity, A
End Conductors

Begin Boundary Conditions
! type    parameter node
  fixed_T   25.0     wall    ! surrounding temperature
  fixed_T  200.0     surf    ! pipe surface
End Boundary Conditions
```

TNSolver Output File

Convection and Surface Radiation Example

```
**************************************************************
*                                                            *
*          TNSolver - A Thermal Network Solver               *
*                                                            *
*              Version 0.9.2, August 9, 2017                 *
*                                                            *
**************************************************************

*** Solution Parameters ***

Title: Example 1.2, p. 10 in [BLID11]

Type                          =  steady
Units                         =  SI
Temperature units             =  C
Nonlinear convergence         =  1e-009
Maximum nonlinear iterations  =  100
Gravity                       =  9.80665 (m/s^2)
Stefan-Boltzmann constant     =  5.67037e-008 (W/m^2-K^4)
```

TNSolver Output File (continued)

Convection and Surface Radiation Example

```
*** Nodes ***

Volume      Temperature
Label     Material    (m^3)      (C)
--------- ---------- ---------- -----------
surf        N/A         0          200
wall        N/A         0           25

*** Conductors ***

                                              Q_ij
   Label         Type        Node i     Node j     (W)
---------- ------------- ---------- ---------- ----------
   conv     convection      surf       wall      577.5
    rad      surfrad        surf       wall     421.311
```

TNSolver Output File (continued)

Convection and Surface Radiation Example

```
*** Boundary Conditions ***

Type        Parameter(s)        Node(s)
----------  ------------------  --------------------
fixed_T         25              wall
fixed_T         200             surf

*** Conductor Parameters ***

surfrad: Surface Radiation

h_r
label    (W/m^2-K)
----------  ----------
rad    10.9431
```

Conclusion

An introduction to thermal network analysis with TNSolver for steady heat conduction, convection and radiation.

Questions?

Appendix

Obtaining GNU Octave
GNU Octave

- ▶ GNU Octave
    - ▶ `http://www.gnu.org/software/octave/`
- ▶ Octave Wiki
    - ▶ `http://wiki.octave.org`
- ▶ Octave-Forge Packages (similar to MATLAB Toolbox packages)
    - ▶ `http://octave.sourceforge.net`

SI 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	$^\circ C = K - 273.15$
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	$\frac{M}{L \cdot t}$	$\frac{kg}{m \cdot s}$	$Pa \cdot s$, $\frac{N \cdot s}{m^2}$
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	$\frac{M}{t^3 \cdot T}$	$\frac{kg}{s^3 \cdot K}$	$\frac{W}{m^2 \cdot K}$, $\frac{J}{s \cdot m^2 \cdot K}$

Cartesian Tensor Notation (Einstein Convention)

Cartesian tensor notation is a compact method for writing equations. A few simple rules can be used to expand an equation into is full form based on the subscript indices. The range of the indices are based on the spatial dimension of the problem. If an index is repeated within a term of the equation, then a summation over the index is implied.

Two-dimensions:

$$q_i n_i = q_1 n_1 + q_2 n_2 = q_x n_x + q_y n_y$$

Three-dimensions:

$$q_i n_i = q_1 n_1 + q_2 n_2 + q_3 n_3 = q_x n_x + q_y n_y + q_z n_z$$

References I

[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.

[LL16]   J. H. Lienhard, IV and J. H. Lienhard, V.
          A Heat Transfer Textbook.
          Phlogiston Press, Cambridge, Massachusetts, fourth
          edition, 2016.
          Available at: http://ahtt.mit.edu.