Federico Giai Pron

District heating system circuit simulator

Solver description

Giai Pron, Federico
6-27-2022

Contents

Figures

Symbols

Variables

Symbol	Name	Unit	Unit symbol
$NumNodes$	Number of nodes	Number	1
$NumElems$	Number of elements	Number	1
$NumIters$	Number of iterations	Number	1
x	x-axis coordinate	Meter	m
y	y-axis coordinate	Meter	m
z	z-axis coordinate	Meter	m
L	Length	Meter	m
D	Diameter	Meter	m
s	Thickness	Meter	m
A	Area	Meter^2	m^2
ϵ	Roughness	Meter	m
p	Pressure	Pascal	Pa
Δp	Pressure difference	Pascal	Pa
T	Temperature	Kelvin	K
X	Global variable	Pascal - Kelvin	Pa - K
u	Speed	Meter / seconds	m/s
G	Mass flow rate	Kilogram / seconds	kg/s
ϕ	Heat flow rate	Watt	W
ρ	Density	Kilogram / meter^3	kg/m^3
μ	Viscosity	Pascal · second	Pa·s
λ	Thermal conductivity	Watt / meter / Kelvin	W/m/K
c_p	Specific heat capacity	Joule / kilogram / Kelvin	J/kg/K
h	Specific enthalpy	Joule / kilogram	J/kg
f	Darcy friction factor	Number	1
Re	Reynolds number	Number	1
Nu	Nusselt number	Number	1
Pr	Prandtl number	Number	1
U	Heat exchange coefficient	Watt / meter^2 / Kelvin	W/m^2/K
H	Variable variation	Variable	Variable
R	Residuals	Variable	Variable
J	Jacobian	Variable	Variable
$d\ldots$	Variation	Variable	Variable
$RelErr$	Relative error	Number	1
k	Iterator	Number	1
P	Power	Watt	W
η	Efficiency	Number	1

Subscripts

Symbol	Name
n	Node
e	Element
g	Global
$n_1(e)$	First node of the element e
$n_2(e)$	Second node of the element e
t	Transversal
l	Longitudinal
src	Source
cv	Convection
cd	Conduction
l	Liquid

s	Solid
ref	Reference
dn	Down
up	Up
$Iters$	Iterations
in	Input
d	Dissipated
p	Pumping
u	Useful
f	Feed

Superscripts

Symbol	Name
ext	External environment
old	Previous iteration

Tool structure

Sign conventions

The adopted sign conventions are shown in the following picture:

- G_e, $\phi_{e,cv}$ and ϕ_e are positive when the mass / heat flow exits from the first node of the element and enters in the second node of the element
- $\phi_{e,cd}$, $\phi_{e,src}$, G_n and ϕ_n are positive when the mass / heat flow exits from the node / element

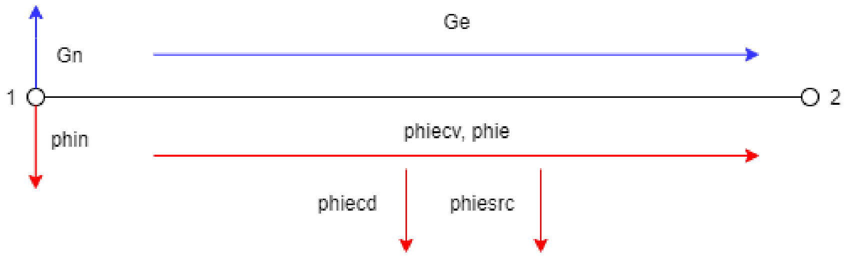

Figure 1 - Sign conventions.

Solver description

File structure

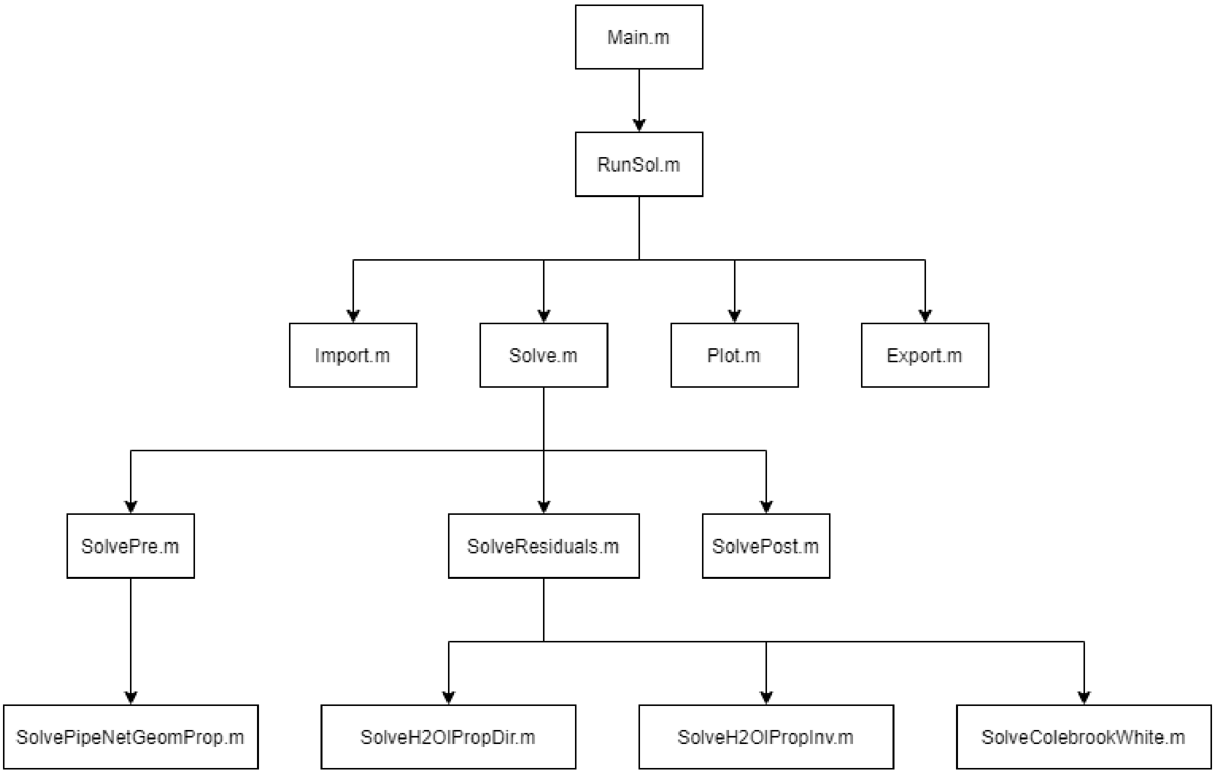

Figure 2 - Tool structure

File: SolvePipeNetGeomProp.m

In this function, the geometrical properties of the pipes network are calculated from nodes coordinates, $Nodes$, pipes network topology, $Elems$, and pipes diameters, D_e.

Determine sizes

In this section, the number of elements is determined:

- $NumElems$

Calculate the geometrical properties of the pipes network

For each element, e:

- Calculate the element length, L_e:
  - $L_e = \sqrt{\left(x_{n_2(e)} - x_{n_1(e)}\right)^2 + \left(y_{n_2(e)} - y_{n_1(e)}\right)^2 + \left(z_{n_2(e)} - z_{n_1(e)}\right)^2}$
- Calculate the element transversal area, $A_{e,t}$:
  - $A_{e,t} = \frac{\pi D_e^2}{4}$
- Calculate the element longitudinal area, $A_{e,l}$:
  - $A_{e,l} = \pi D_e L_e$

File: SolveH2OlPropDir.m

In this function, the properties of the liquid water are calculated as a function of the temperature, T.

- Calculate density, ρ_l, viscosity, μ_l, thermal conductivity, λ_l, specific heat capacity, $c_{p,l}$, and specific enthalpy, h_l, as a function of the temperature, T:
  - $\left[\rho_l, \mu_l, \lambda_l, c_{p,l}, h_l\right] = SolveH2OlPropDir(T)$

File: SolveH2OlPropInv.m

In this function, the temperature of the liquid water, T, is calculated as a function of the specific heat capacity, $c_{p,l}$, and of the specific enthalpy, h_l:

- Define the reference temperature, T_{ref};
- Calculate the temperature, T:
  - $T = T_{ref} + \frac{h_l}{c_{p,l}}$

File: SolveColebrookWhite.m

In this function, the Colebrook-White equation is solved with the Newton-Raphson method.

Calculate the initial guess of the Darcy friction factor

- Calculate the initial guess of the Darcy friction factor, f_e:
  - $f_e = \left\{-2\log\left[\frac{\epsilon_e}{3.7} - \frac{5.02}{Re_e}\log\left(\epsilon_e - \frac{5.02}{Re_e}\log\left(\frac{\epsilon_e}{3.7} + \frac{13}{Re_e}\right)\right)\right]\right\}^{-2}$

  Reference: equation 12 by (Genić, et al., 2011).

Apply the Newton-Raphson method

- Define Newton-Raphson method parameters:
  - Down / up variables variation, H;
  - Number of iterations, $NumIters$;

For each iteration, k_{Iters}:

- Calculate down / up variables, $f_{e,dn}$ and $f_{e,up}$:
  - $f_{e,dn} = f_e - \frac{H}{2}$
  - $f_{e,up} = f_e + \frac{H}{2}$
- Calculate the Jacobian matrix, J_e:
  - Calculate down / up residual vectors, $R_{e,dn}$ and $R_{e,up}$:
    - $R_{e,dn} = \frac{1}{\sqrt{f_{e,dn}}} + 2\log\left(\frac{\epsilon_e}{3.7\,D_e} + \frac{2.51}{Re_e\sqrt{f_{e,dn}}}\right)$
    - $R_{e,up} = \frac{1}{\sqrt{f_{e,up}}} + 2\log\left(\frac{\epsilon_e}{3.7\,D_e} + \frac{2.51}{Re_e\sqrt{f_{e,up}}}\right)$
  - Calculate the Jacobian matrix, J_e:
    - $J_e = \frac{R_{e,up} - R_{e,dn}}{H}$
- Calculate the residual vector, R_e:
  - $R_e = \frac{1}{\sqrt{f_e}} + 2\log\left(\frac{\epsilon_e}{3.7\,D_e} + \frac{2.51}{Re_e\sqrt{f_e}}\right)$
- Calculate the correction, df_e:
  - $df_e = -\frac{R_e}{J_e}$
- Update variable, f_e:
  - $f_e = f_e + df_e$
- Calculate the relative error, $RelErr$:
  - $RelErr = |R_e|$

File: SolvePre.m

In this section, the data pre-processing is performed.

Determine sizes

In this section, the number of nodes and elements are determined:

- $NumNodes$
- $NumElems$

Define parameters
In this section, the following parameters are defined:

- External pressure, p_n^{ext}
- External temperature, T_n^{ext}

Calculate the geometrical properties of the pipes network
In this section, the geometrical properties of the pipes network are calculated from nodes coordinates, $Nodes$, pipes network topology, $Elems$, and pipes diameters, D_e:

- $[L_e, A_{e,t}, A_{e,l}] = SolvePipeNetGeomProp(Nodes, Elems, D_e)$

Elaborate inputs
In this section, the structure of the inputs is converted into vectorial form:

- $p_{n,src,bc} \rightarrow p_{n,src}$
- $T_{n,src,bc} \rightarrow T_{n,src}$
- $G_{n,src,bc} \rightarrow G_{n,src}$
- $\phi_{n,src,bc} \rightarrow \phi_{n,src}$
- $\phi_{e,src,bc} \rightarrow \phi_{e,src}$

Calculate sources properties
In this section, the thermal-fluid-dynamic properties of the sources are calculated:

- $[\rho_{n,l,src}, \mu_{n,l,src}, \lambda_{n,l,src}, c_{p,n,l,src}, h_{n,l,src}] = SolveH2OPropDir(T_{n,src})$

Determine input / output source mass / heat flow rates
In this section, the input / output source mass / heat flow rates are determined:

- $G_{n,src,out} = \sum_n^{NumNodes} \begin{cases} |G_{n,src}|, & G_{n,src} > 0 \\ 0, & G_{n,src} < 0 \end{cases}$ and $G_{n,src,in} = \sum_n^{NumNodes} \begin{cases} 0, & G_{n,src} > 0 \\ |G_{n,src}|, & G_{n,src} < 0 \end{cases}$
- $\phi_{n,src,out} = \sum_n^{NumNodes} \begin{cases} |\phi_{n,src}|, & \phi_{n,src} > 0 \\ 0, & \phi_{n,src} < 0 \end{cases}$ and $\phi_{n,src,in} = \sum_n^{NumNodes} \begin{cases} 0, & \phi_{n,src} > 0 \\ |\phi_{n,src}|, & \phi_{n,src} < 0 \end{cases}$
- $\phi_{e,src,out} = \sum_e^{NumElems} \begin{cases} |\phi_{e,src}|, & \phi_{e,src} > 0 \\ 0, & \phi_{e,src} < 0 \end{cases}$ and $\phi_{e,src,in} = \sum_e^{NumElems} \begin{cases} 0, & \phi_{e,src} > 0 \\ |\phi_{e,src}|, & \phi_{e,src} < 0 \end{cases}$

File: SolveResiduals.m
In this function, the model is implemented.

Determine sizes
In this section, the number of nodes and elements are determined:

- $NumNodes$
- $NumElems$

Define parameters
In this section, the following parameters are defined:

- External temperature, T_e^{ext}

Elaborate inputs
In this section:

- The node pressures and temperatures are extracted from the variable vector of the Newton-Raphson method:
  - $X_n = \begin{bmatrix} p_n \\ T_n \end{bmatrix} \rightarrow p_n, T_n$

Calculate fluid properties in correspondence of each node

In this section, the fluid properties in correspondence of each node are calculated:

- $[\rho_{n,l}, \mu_{n,l}, \lambda_{n,l}, c_{p,n,l}, h_{n,l}] = SolveH2OPropDir(T_n)$

Calculate elements variables

For each element e:

- Calculate the average temperature along the element e, T_e, from the temperature of the element nodes, $T_{n_1(e)}$ and $T_{n_2(e)}$:
  - $T_e = \frac{T_{n_1(e)}+T_{n_2(e)}}{2}$
- Calculate the average fluid properties along the element e, $\rho_{e,l}$, $\mu_{e,l}$, $\lambda_{e,l}$ and $c_{p,e,l}$, from the average temperature along the element e, T_e:
  - $[\rho_{e,l}, \mu_{e,l}, \lambda_{e,l}, c_{p,e,l}] = Solve_{H_2O\ prop,Dir}(T_e)$
- Calculate the "old" speed along the element e, u_e^{old}, from the "old" mass flow rate along the element e, G_e^{old}:
  - $u_e^{old} = \frac{G_e^{old}}{\rho_{e,l} A_{e,t}}$
- Calculate the "old" Reynolds number along the element e, Re_e^{old}, from the "old" speed along the element e, u_e^{old}:
  - $Re_e^{old} = \frac{\rho_{e,l}|u_e^{old}|D_e}{\mu_{e,l}}$
- Calculate the "old" Darcy friction factor along the element e, f_e^{old}, as:
  - $f_e^{old} = \begin{cases} \frac{64}{Re_e^{old}}, & Re_e^{old} < 2300 \\ SolveColebrookWhite(De, \epsilon_e, Re_e^{old}), & Re_e^{old} \geq 2300 \end{cases}$
- Assume the Darcy friction factor along the element e, f_e, to be equal to the "old" Darcy friction factor along the element e, f_e^{old}:
  - $f_e = f_e^{old}$
- Calculate the pressure difference along the element e, Δp_e, from the pressure of the element nodes, $p_{n_1(e)}$ and $p_{n_2(e)}$:
  - $\Delta p_e = p_{n_1(e)} - p_{n_2(e)}$
- Calculate the speed along the element e, u_e, through the Darcy-Weisbach equation (i.e. $\Delta p_e = f_e \frac{L_e}{D_e} \rho_{e,l} \frac{u_e^2}{2}$):
  - $u_e = sign(\Delta p_e) \sqrt{\frac{2\,|\Delta p_e|}{f_e \frac{L_e}{D_e} \rho_{e,l}}}$
- Calculate the mass flow rate along the element e, G_e:
  - $G_e = \rho_{e,l} u_e A_{e,t}$
- Calculate the Reynolds, Prandtl and Nusselt numbers along the element e, Re_e, Pr_e and Nu_e:
  - $Re_e = \frac{\rho_{e,l}|u_e|D_e}{\mu_{e,l}}$
  - $Pr_e = \frac{\mu_{e,l} c_{p,e,l}}{\lambda_{e,l}}$
  - $Nu_e = 0.0265\, Re_e^{0.8} Pr_e^{0.3}$
- Calculate the convective heat flow rate along the element e, $\phi_{e,cv}$:
  - $\phi_{e,cv} = \begin{cases} G_e h_{n_1(e),l}, & G_e > 0 \\ G_e h_{n_2(e),l}, & G_e < 0 \end{cases}$
- Calculate the conductive heat transfer coefficient along the element e, $U_{e,cd}$:
  - $U_{e,cd,l} = \frac{Nu_e \lambda_{e,l}}{D_e}$
  - $U_{e,cd,s} = \frac{\lambda_{e,s} A_{e,l}}{s_e}$
  - $U_{e,cd} = \frac{1}{\frac{1}{U_{e,cd,s}} + \frac{1}{U_{e,cd,l}}}$
- Calculate the conductive heat flow rate along the element e, $\phi_{cd,e}$:

- $\phi_{e,cd} = U_{e,cd}A_{e,l}(T_e - T_e^{ext})$

- Calculate the total heat flow rate along the element e, ϕ_e:
  - $\phi_e = \begin{cases} \max(0, \phi_{e,cv} - \phi_{e,cd} - \phi_{e,src}), & G_e > 0 \\ \min(0, \phi_{e,cv} + \phi_{e,cd} + \phi_{e,src}), & G_e < 0 \end{cases}$

Calculate nodes variables

For each element e:

- Calculate the net mass flow rate in node n, G_n:
  - $G_{n_1(e)} = G_{n_1(e)} - G_e$
  - $G_{n_2(e)} = G_{n_2(e)} + G_e$
- Calculate the net heat flow rate in node n, ϕ_n:
  - $\phi_{n_1(e)} = \phi_{n_1(e)} - \phi_e$
  - $\phi_{n_2(e)} = \phi_{n_2(e)} + \phi_e$

For each node n:

- Add the contribute of mass source in node n, G_n:
  - $\phi_n = \begin{cases} \phi_n + G_n h_{n,l}, & G_n > 0 \\ \phi_n + G_n h_{n,l,src}, & G_n < 0 \end{cases}$

  N.B. It has not been considered $G_{n,src}$ because in case a pressure is imposed on node n, $G_n \neq G_{n,src}$
- Add the contribute of heat source in node n, $\phi_{n,src}$:
  - $\phi_n = \phi_n + \phi_{n,src}$

Calculate pressure residuals

- Calculate pressure residuals, R_{p_n}, as the difference of the net mass flow rate in node n, G_n, and the source mass flow rate at node n, $G_{n,src}$:
  - $R_{p_n} = G_n - G_{n,src}$

Calculate temperature residuals

- Calculate the input mass flow rate in node n, $G_{n,in}$:
  For each element e:
  - $\begin{cases} G_{n_2(e),in} = G_{n_2(e),in} + |G_e|, & G_e > 0 \\ G_{n_1(e),in} = G_{n_1(e),in} + |G_e|, & G_e < 0 \end{cases}$

  For each node n:
  - $G_{n,in} = G_{n,in} + \begin{cases} 0, & G_n > 0 \\ |G_n|, & G_n < 0 \end{cases}$
- Calculate the input heat flow rate in node n, $\phi_{n,in}$:
  For each element e:
  - $\begin{cases} \phi_{n_2(e),in} = \phi_{n_2(e),in} + |\phi_e|, & \phi_e > 0 \\ \phi_{n_1(e),in} = \phi_{n_1(e),in} + |\phi_e|, & \phi_e < 0 \end{cases}$

  For each node n:
  - $\phi_{n,in} = \phi_{n,in} + \begin{cases} 0, & G_n > 0 \\ |G_n| h_{n,l,src}, & G_n < 0 \end{cases} - \phi_{n,src}$

    A negative sign is applied to $\phi_{n,src}$ because $\phi_{n,in}$ is considered positive when entering while $\phi_{n,src}$ when exiting.

- Calculate the input enthalpy in node n, $h_{n,l,in}$:
  - $h_{n,l,in} = \frac{\phi_{n,in}}{G_{n,in}}$
- Calculate the input temperature in node n, $T_{n,in}$:
  - $T_{n,in} = SolveH2OPropInv(c_{p,n,l}, h_{n,l,in})$
- Calculate the temperature residuals, R_{T_n}, as the difference of the temperature in node n, and the input temperature in node n, $T_{n,in}$:

- $R_{T_n} = T_n - T_{n,in}$

Determine residuals

- Unify pressure and temperature residuals to get a unique residual vector:
  - $R_n = \begin{bmatrix} R_{p_n} \\ R_{T_n} \end{bmatrix}$

File: SolvePost.m

In this section, the data post-processing is performed.

Determine sizes

In this section, the number of nodes and elements are determined:

- $NumNodes$
- $NumElems$

Calculate the pumping losses

In this section, the pumping losses are calculated:

- Calculate the element pumping losses, $P_{e,d,p}$:
  - $P_{e,d,p} = |\Delta p_e A_{e,t} u_e|$
- Calculate the global pumping losses, $P_{g,d,p}$:
  - $P_{g,d,p} = \sum_e^{NumElems} P_{e,d,p}$

Calculate the thermal losses

In this section, the thermal losses are calculated:

- Calculate the element thermal losses, $P_{e,d,t}$:
  - $P_{e,d,t} = |\phi_{e,cd}|$
- Calculate the global thermal losses, $P_{g,d,t}$:
  - $P_{g,d,t} = \sum_e^{NumElems} P_{e,d,t}$

Calculate the total losses

In this section, the total losses are calculated:

- Calculate the element total losses, $P_{e,d}$:
  - $P_{e,d} = P_{e,d,p} + P_{e,d,t}$
- Calculate the global losses, $P_{g,d}$:
  - $P_{g,d} = \sum_e^{NumElems} P_{e,d}$

Calculate the useful power

In this section, the useful power is calculated:

- Calculate the node useful power, $P_{n,u}$:
  - $P_{n,u} = \sum_{k_{G_{n,src,bc}}} \begin{cases} 0, & G_{n,src,bc} < 0 \\ G_{n,src,bc} h_{n,l}, & G_{n,src,bc} > 0 \end{cases} + \sum_{k_{\phi_{n,src,bc}}} \begin{cases} 0, & \phi_{n,src,bc} < 0 \\ \phi_{n,src,bc}, & \phi_{n,src,bc} > 0 \end{cases}$
- Calculate the element useful power, $P_{e,u}$:
  - $P_{e,u} = \sum_{k_{\phi_{e,src,bc}}} \begin{cases} 0, & \phi_{e,src,bc} < 0 \\ \phi_{n,src,bc}, & \phi_{e,src,bc} > 0 \end{cases}$
- Calculate the global useful power:
  - $P_{g,u} = \sum_{k_{G_{n,src,bc}}} \begin{cases} 0, & G_{n,src,bc} < 0 \\ G_{n,src,bc} h_{n,l}, & G_{n,src,bc} > 0 \end{cases} + \sum_{k_{\phi_{n,src,bc}}} \begin{cases} 0, & \phi_{n,src,bc} < 0 \\ \phi_{n,src,bc}, & \phi_{n,src,bc} > 0 \end{cases} +$
    $+ \sum_{k_{\phi_{e,src,bc}}} \begin{cases} 0, & \phi_{e,src,bc} < 0 \\ \phi_{n,src,bc}, & \phi_{e,src,bc} > 0 \end{cases}$

Calculate the feed power

In this section, the feed power is calculated:

- Calculate the global feed power:
  - $P_{g,f} = P_{g,u} + P_{g,d}$

Calculate the efficiency

- Calculate the efficiency, η:
  - $\eta = \frac{1}{1+\frac{P_{g,d}}{P_{g,u}}}$

Calculate the element pressure and Darcy friction factor

- Calculate the average pressure along the element e, p_e, from the pressure of the element nodes, $p_{n_1(e)}$ and $p_{n_2(e)}$:
  - $p_e = \frac{p_{n_1(e)}+p_{n_2(e)}}{2}$
- Calculate the "new" Darcy friction factor along the element e, f_e, as:
  - $f_e = \begin{cases} \frac{64}{Re_e}, & Re_e < 2300 \\ SolveColebrookWhite(De, \epsilon_e, Re_e), & Re_e \geq 2300 \end{cases}$

Calculate the input mass / heat flow rate

- Calculate the output mass flow rate in node n, $G_{n,out}$:
  For each element e:
  - $\begin{cases} G_{n_2(e),out} = G_{n_2(e),out} + |G_e|, & G_e < 0 \\ G_{n_1(e),out} = G_{n_1(e),out} + |G_e|, & G_e > 0 \end{cases}$

  For each node n:
  - $G_{n,out} = G_{n,out} + \begin{cases} 0, & G_n < 0 \\ |G_n|, & G_n > 0 \end{cases}$
- Calculate the output heat flow rate in node n, $\phi_{n,out}$:
  For each element e:
  - $\begin{cases} \phi_{n_2(e),out} = \phi_{n_2(e),out} + |\phi_e|, & \phi_e < 0 \\ \phi_{n_1(e),out} = \phi_{n_1(e),out} + |\phi_e|, & \phi_e > 0 \end{cases}$

  For each node n:
  - $\phi_{n,out} = \phi_{n,out} + \begin{cases} 0, & G_n < 0 \\ |G_n| h_{n,l,src}, & G_n > 0 \end{cases} - 0 \cdot \phi_{n,src}$

    A negative sign is applied to $\phi_{n,src}$ because $\phi_{n,in}$ is considered positive when entering while $\phi_{n,src}$ when exiting. The 0 is due to the fact that the term $\phi_{n,src}$ has been already considered in the calculation of the input heat flow rate, $\phi_{n,in}$. If it would be considered also here, then it would be counted twice.

- Calculate the input / output element mass / heat flow rates, $G_{e,in}$, $G_{e,out}$, $\phi_{e,cv,in}$, $\phi_{e,cv,out}$, $\phi_{e,cd,in}$, $\phi_{e,cd,out}$, $\phi_{e,in}$ and $\phi_{e,out}$:
  - $\begin{cases} x_{e,in} = |x_e|, & x_e < 0 \\ x_{e,out} = |x_e|, & x_e > 0 \end{cases}$

  Note: in the previous equation, the symbol x is used to generalize the formula which is the same for all the variables cited above.

File: Solve.m

Determine sizes

In this section, the number of nodes and elements are determined:

- $NumNodes$
- $NumElems$

Run pre-processor

In this section, the data pre-processing is performed.

- $[L_e, A_{e,t}, A_{e,l}, p_{n,src}, T_{n,src}, \rho_{n,l,src}, \mu_{n,l,src}, \lambda_{n,l,src}, c_{p,n,l,src}, h_{n,l,src}, G_{n,src,in}, G_{n,src,out}, G_{n,src}, \phi_{n,src,in}, \phi_{n,src,out},$

$$\phi_{n,src}, \phi_{e,src,in}, \phi_{e,src,out}, \phi_{e,src}] = SolvePre(p_{ext}, T_{ext}, Nodes, Elems, D_e, p_{n,src,bc}, T_{n,src,bc}, G_{n,src,bc}, \phi_{n,src,bc}, \phi_{e,src,bc})$$

Initialize pressure and temperature variables

In this section, pressure and temperature variables are initialized based on boundary conditions:

- Initialize pressure and temperature to the corresponding mean source values:
  - $p_n = mean(p_{n,src,bc})$
  - $T_n = mean(T_{n,src,bc})$
- Add random values to p_n and T_n to guarantee no null Jacobian matrix at the first iteration:
  - $p_n = p_n + \left(-\frac{p_n}{1000} + rand(1)\frac{p_n}{500}\right)$
  - $T_n = T_n + \left(-\frac{T_n}{1000} + rand(1)\frac{T_n}{500}\right)$
- Impose boundary conditions:

  For each pressure boundary condition, $k_{p_{n,src,bc}}$:

  - $p_n[p_{n,src,bc}(k_{p_{n,src,bc}}, 1)] = p_{n,src,bc}(k_{p_{n,src,bc}}, 2)$

  For each temperature boundary condition, $k_{T_{n,src,bc}}$:

  - $T_n[T_{n,src,bc}(k_{T_{n,src,bc}}, 1)] = T_{n,src,bc}(k_{T_{n,src,bc}}, 2)$
- Initialize global variable, X_n:
  - $X_n = \begin{bmatrix} p_n \\ T_n \end{bmatrix}$

Apply the Newton-Raphson method

- Define Newton-Raphson method parameters:
  - Down / up variables variation, H;
  - Number of iterations, Num_{Iters};

For each iteration k_{Iters}:

- Calculate the Jacobian matrix, J_n:
  For each variable $X_n(j)$:
  - Initialize down / up variables, $X_{n,dn}$ and $X_{n,up}$:
    - $X_{n,dn} = X_n$
    - $X_{n,up} = X_n$
  - Calculate down / up variables, $X_{n,dn}$ and $X_{n,up}$:
    - $X_{n,dn}(j) = X_n(j) - \frac{H}{2}$
    - $X_{n,up}(j) = X_n(j) + \frac{H}{2}$
  - Calculate down / up residual vectors, $R_{n,dn}$ and $R_{n,up}$:
    - $R_{n,dn}(:,j) = SolveResiduals(Nodes, Elems, L_e, D_e, s_e, A_{e,t}, A_{e,l}, \epsilon_e, \lambda_{e,s}, p_{n,src,bc}, T_{n,src,bc}, G_{n,src,bc}, \phi_{e,src,bc}, \phi_{n,src,bc}, G_{e,old}, k_{Iters}, X_{n,dn})$
    - $R_{n,up}(:,j) = SolveResiduals(Nodes, Elems, L_e, D_e, s_e, A_{e,t}, A_{e,l}, \epsilon_e, \lambda_{e,s}, p_{n,src,bc}, T_{n,src,bc}, G_{n,src,bc}, \phi_{e,src,bc}, \phi_{n,src,bc}, G_{e,old}, k_{Iters}, X_{n,up})$
  - Calculate the Jacobian matrix, J_n:
    - $J_n(:,j) = \frac{R_{n,up}(:,j) - R_{n,dn}(:,j)}{H}$
- Calculate the residual vector, R_n:
  - $R_n = SolveResiduals(Nodes, Elems, L_e, D_e, s_e, A_{e,t}, A_{e,l}, \epsilon_e, \lambda_{e,s}, p_{n,src,bc}, T_{n,src,bc}, G_{n,src,bc}, \phi_{e,src,bc}, \phi_{n,src,bc}, G_{e,old}, k_{Iters}, X_n)$
- Impose pressure boundary conditions:
  For each pressure boundary condition, $k_{p_{n,src,bc}}$:

  - $J_n[p_{n,src,bc}(k_{p_{n,src,bc}},1),:] = \begin{cases} 1, & Col = p_{n,src,bc}(k_{p_{n,src,bc}},1) \\ 0, & Col \neq p_{n,src,bc}(k_{p_{n,src,bc}},1) \end{cases}$
  - $R_n[p_{n,src,bc}(k_{p_{n,src,bc}},1)] = 0$
- Calculate the correction, dX_n:
  - $dX_n = -\frac{J_n}{R_n}$
- Update variable, X_n:
  - $X_n = X_n + dX_n$
- Impose pressure boundary conditions:
  For each pressure boundary condition, $k_{p_{n,src,bc}}$:
  - $X_n[p_{n,src,bc}(k_{p_{n,src,bc}},1)] = p_{n,src,bc}(k_{p_{n,src,bc}},2)$
- Calculate the relative error, $RelErr$:
  - $RelErr(1) = |norm(R_n(0\ NumNodes + 1:1:1\ NumNodes))|$
  - $RelErr(2) = |norm(R_n(1\ NumNodes + 1:1:2\ NumNodes))|$

After the last iteration:

- Determine pressure and temperature, p_n and T_n, from variable vector, X_n:
  - $X_n \rightarrow p_n, T_n$

Run the post-processor

In this section, the data post-processing is performed.

- $[P_{g,f}, P_{g,d,p}, P_{g,d,t}, P_{g,d}, P_{g,u}, \eta, G_{n,out}, \phi_{n,out}, p_e, f_e, G_{e,in}, G_{e,out}, \phi_{e,cv,in}, \phi_{e,cv,out}, \phi_{e,cd,in}, \phi_{e,cd,out}, \phi_{e,in}, \phi_{e,out}, SolveColebrookRelErrf_e] = SolvePost(SolveColebrookWhiteNumIters, SolveColebrookWhiteHfe, Nodes, p_n, T_n, h_{n,l}, G_n, \phi_n, G_{n,src,bc}, \phi_{n,src,bc}, dp_e, Elems, D_e, \epsilon_e, A_{e,t}, u_e, Re_e, G_e, \phi_{e,cv}, \phi_{e,cd}, \phi_e, \phi_{e,src,bc})$

Bibliography

Genić, S., Arandjelović, I., Kolendić, P., Jarić, M., Budimir, N., & Genić, V. (2011). A Review of Explicit Approximations of Colebrook's Equation. *FME Transactions*, 67-71.

Lindgren, J. (2015). *Numerical modelling of district heating networks.*

Sarbu, I., Mirza, M., & Crasmareanu, E. (2019). A review of modelling and optimisation techniques for district heating systems. *Wiley - International journal of energy research*, 6572-6598.

Tol, H. I. (2021). Development of a physical hydraulic modelling tool for District Heating systems. *Energy & Buildings*.

Zheng, X., Sun, Q., Wang, Y., Zheng, L., Gao, X., You, S., . . . Shi, K. (2021). Thermo-hydraulic coupled simulation and analysis of a real large-scale complex district heating network in Tianjin. *Energy*.