ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ
V PRAZE
FAKULTA STROJNÍ
12112 – Ústav mechaniky tekutin a termodynamiky

Development of Mathematical Model of Compact Type Evaporator for Air Dehumidification

Bc. Martin Borovička

2017

Vedoucí práce: Ing. Tomáš Hyhlík, Ph.D.

Oficiální zadání

Čestné prohlášení

Prohlašuji, že jsem svou diplomovou práci vypracoval samostatně a použil jsem pouze podklady uvedené v přiloženém seznamu.

Nemám závažný důvod proti použití tohoto školního díla ve smyslu § 60 zákona č. 121/2000Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Praze dne …… ……

podpis

Poděkování

Děkuji panu Ing. Tomáši Hyhlíkovi, Ph.D. za velmi vstřícný přístup při vedení této diplomové práce a za čas, který mi věnoval při konzultacích. Dále děkuji Ing. Lukáši Dvořákovi, Ph.D., Ing. Zdeňku Sumarovi a Bc. Michalu Šochmanovi za pomoc při stavbě testovacího zařízení a provedení experimentů nezbytných k validaci modelu. Velké poděkování patří také mé rodině za poskytnuté zázemí a podporu při studiu.

Acknowledgement

I would first like to thank my thesis advisor Ing. Tomáš Hyhlík, Ph.D for his valuable comments on this thesis. I would also like to acknowledge Ing. Lukáš Dvořák, Ph.D., Ing. Zdeňek Sumara and Bc. Michal Šochman for assistance during test rig assembly and conduction of necessary experiments. Finally, I must express my very profound gratitude to my family for providing me with unfailing support and encouragement.

Anotace

Jméno autora:	Bc. Martin Borovička
Instituce:	České Vysoké Učení Technické v Praze
	Fakulta strojní
	12112 – Ústav mechaniky tekutin a termodynamiky
	Technická 4, 166 07, Praha 6

Obor:	Aplikovaná mechanika
Název práce:	Vývoj matematického modelu výparníku kompaktního typu pro odvlhčování vlhkého vzduchu
Vedoucí práce:	Ing. Tomáš Hyhlík, Ph.D.
Počet stran:	78
Rok:	2016/2017
Klíčová slova:	matematický model, výparník, vlhký vzduch, odvlhčování

Diplomová práce se zabývá vývojem matematického modelu pro výparníky kompaktního typu – trubkový výměník s lamelami, určené pro odvlhčování vlhkého vzduchu. V úvodu práce jsou uvedeny standartní přístupy k analýze tepelných výměníků a několik příkladů ad-hoc numerických modelů pro řešení přestupu tepla a hmoty ve výměnících a chladících věží. V další části jsou popsány požadavky na vyvíjený MATLAB model, jeho pod-procedury a řešiče, které řeší různé problémy z oblasti přenosu tepla a hmoty – povrchová teplota desky, teplota vodního filmu stékajícího po stěně, lokální parametry proudu vlhkého vzduchu a podmínky pro kondenzaci na stěně a rozložení teploty chladiva v trubkovém svazku. Model je následně pro vybraný výměník v několika fázích validován. Experimentálních data, použitá k validaci, jsou získána měřením daného výměníku na testovací trati. Za účelem použití výsledků modelu v rozsáhlejší CFD simulaci, je vytvořen automatický postup přenosu dat z MATLAB modelu do CFD kódu (STAR-CCM+), za pomoci Java/C skriptů a procedur. V práci je také obsažen krátký uživatelský manuál, který pomůže uživateli nastavit počáteční podmínky, geometrii výměníku a také řídit samotný proces výpočtu, parametry řešičů a kontrolovat tak stabilitu řešení. Do MATLAB modelu je následně naimplementována metoda Proper Orthogonal Decomposition (POD), která umožňuje rekonstrukci řešení pro nesimulované vstupní parametry a tím tedy redukci množství dat pro přesun do CFD softwaru.

Abstract

Author's name:	Bc. Martin Borovička
Institution:	Czech Technical University in Prague
	Faculty of Mechanical Engineering
	12112 – Department of Fluid Dynamics and Thermodynamics
	Technická 4, 166 07, Praha 6
Programme:	Applied mechanics
Title:	Development of Mathematical Model of Compact Type Evaporator for Air Dehumidification
Consultant:	Ing. Tomáš Hyhlík, Ph.D.
Number of pages:	78
Year:	2016/2017
Keywords:	mathematical model, evaporator, humid air, dehumidification,

This master's thesis deals with the mathematical model development for compact type evaporator, with focus on air dehumidification application Standard methods for heat exchanger analysis and examples of ad-hoc numerical models for solving heat and mass transfer problems in cooling towers and heat exchangers are stated in the first part of the thesis. Requirements and properties of mathematical model developed in MATLAB software are presented in next part, along with model sub-procedures and solvers, that deal with various heat and mass transfer problems – fin surface temperature, condensate film temperature, local humid air parameters and conditions for condensation and refrigerant temperature distribution in tube bundle. The model is then validated in multiple stages for selected heat exchanger geometry. An automatic procedure of data transfer is developed in order to use results of MATLAB model in more complex simulation within commercial CFD code (STAR-CCM+). User manual, that helps user to setup inlet conditions, evaporator geometry, solver parameters and control a solution stability is included in this work. In the end, Proper Orthogonal Decomposition (POD) method is introduced and implemented into MATLAB model. This method helps to reduce the amount of data for transfer into CFD, because it allows to reconstruct non-simulated solution.

Table of content

Nomenclature

Used symbols

Symbol	Unit	Name - description
c_p	$\mathrm{J \cdot kg^{-1} \cdot K^{-1}}$	specific heat/heat capacity at constant pressure
l_{230}	$\mathrm{J \cdot kg^{-1}}$	latent heat of evaporation
Δh_{23}	$\mathrm{J \cdot kg^{-1}}$	enthalpy of evaporation
k	$\mathrm{W \cdot m^{-2} \cdot K^{-1}}$	overall heat transfer coefficient
f	1	friction factor
$\dot{m}$	$\mathrm{kg \cdot s^{-1}}$	mass flow rate
$\dot{m}''$	$\mathrm{kg \cdot m^{-2} \cdot s^{-1}}$	mass flux
w	$\mathrm{kg_{H2O}/kg_{ha}}$	mass fraction
$\boldsymbol{q}$	$\mathrm{W \cdot m^{-2}}$	heat flux vector
$\boldsymbol{D}$	$\mathrm{W \cdot m^{-1} \cdot K^{-1}}$	thermal conductivity matrix
$\boldsymbol{\nabla} T$	$\mathrm{K \cdot m^{-1}}$	temperature gradient
x	$\mathrm{kg_{H2O}/kg_{da}}$	specific humidity
D	$\mathrm{m^2 \cdot s^{-1}}$	diffusivity
p	Pa	pressure
p_0	Pa	barometric pressure
p_v	Pa	water vapor saturation pressure
T	°C, K	temperature
β	$\mathrm{m \cdot s^{-1}}$	mass transfer coefficient
s	mm, m	fin/plate thickness
δ	mm, m	condensate film thickness
h	$\mathrm{J \cdot kg^{-1}}$	enthalpy
$\dot{H}$	W	enthalpy flow
δT_j^i	1	iteration temperature residual
δw_v^i	1	iteration humidity residual
q_{gw}	$\mathrm{W \cdot m^{-2}}$	heat flux from air to fin/film
ε	1	effectiveness in ε-NTU method
α	$\mathrm{W \cdot m^{-2} \cdot s^{-1}}$	heat transfer coefficient
L_x	mm, m	evaporator length
L_y	mm, m	evaporator height
L_{ztot}	mm, m	evaporator width
φ	1	relative humidity
Re	1	Reynolds number
Nu	1	Nusselt number

Pr	1	Prandtl number
Sh	1	Sherwood number
Sc	1	Schmidt number
Le	1	Lewis number
Le_f	1	Lewis factor
$KMit$		number of iterations of solution
NTU	1	number of transfer units
$LMTD$	1	logarithmic mean temp. difference
Nd	1	number of fins in heat exchanger
$Telem$	1	number of elements in tube cross-section
$ErrSurf$	%	error of tube section area
N	1	number of elements in x direction
M	1	number of elements in y direction
$nTPR$	1	number of tubes per row in tube bundle
$nRows$	1	number of tube rows in tube bundle
HB_{err}	%	error of heat balance
$\overline{\boldsymbol{T}}$	°C, K	average temperature
$\overline{\boldsymbol{w}}$	$\mathrm{kg_{H2O}/kg_{ha}}$	average humidity/average mass fraciton
q_b, q_n	$\mathrm{W \cdot m^{-2}}$	boundary heat flows
Q	$\mathrm{W \cdot m^{-3}}$	internal heat supply
Ω	-	area of elements
$\Omega_{q,T}$	-	elements boundary with heat or temperature condition
μ	$\mathrm{Pa \cdot s}$	dynamic viscosity
ν	$m^2 \cdot s^{-1}$	kinematic viscosity
τ_g	Pa	shear stress at air-film interface
E_T	1	Ackermann correction factor
R	$\mathrm{J \cdot mol^{-1} \cdot K^{-1}}$	gas constant
λ	$\mathrm{W \cdot m^{-1} \cdot K^{-1}}$	heat/thermal conductivity
F	1	correction factor for MLTD
y	1	mole fraction
$\tilde{y}$	1	mole fraction at saturation
u	$m \cdot s^{-1}$	velocity

Used subscripts

a	air
da	dry air
ha	humid air
$H2O$	water
w	wall (fin or tube surface)
v	water vapor
$f/film$	condensate film
fw	film – wall interface
mix	air with water vapor
in	inlet
out	outlet
r	refrigerant
c	coolant
l	liquid
SAT	saturated state

Used abbreviations

HT	heat transfer
MT	mass transfer
URF	under-relaxation factor
CS	system of coordinates
HB	heat balance
HD	heat duty
RCS	redistribution correction solver
SW	software
HE	heat exchanger
HTC	heat transfer coefficient
MTC	mass transfer coefficient
UDF	user defined function
UCF	user coded function
FEM	finite element method
RH	relative humidity
$LMTD$	logarithmic mean temperature difference

1 Introduction and main objective

An increasing effort to decrease the cost of any device requires more advanced methods for precise simulation and modeling of physical phenomena. Heat and mass transfer in heat exchangers/evaporator with complex geometry is still a complicated problem, because an analytical solution is not often available, so numerical simulation along with measurement is the only option that can be used for heat duty estimation and geometry optimization. This master's thesis is focused on modeling of heat and mass transfer inside evaporators for air dehumidification. Selected evaporator for this application is classified as a cross flow compact type heat exchanger (or finned tube heat exchanger) - tube bundle (single tube with multiple passes) with common fins (every rectangular fin is in contact with all passes of the tube), using R134a as a refrigerant. An evaporator is placed in heat pump circuit that can be often found in many air dehumidification devices. Other parts of heat pump circuits are: condenser, compressor and capillary tube/reduction valve. All these parts are designed as a unit, so optimization of one part is not very effective approach, optimization of the whole circuit is required.

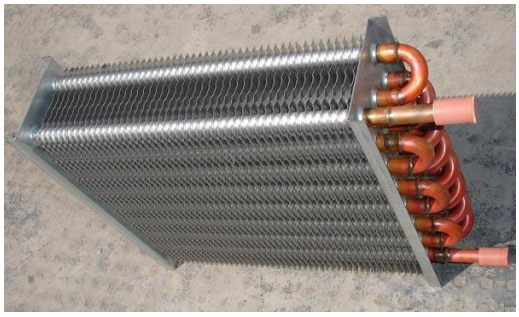


Fig. 1 Compact type – finned tube heat exchanger [1]

The main objective of the master's thesis is to develop parametric a mathematical model for this type of evaporators that is capable of simulating the total heat duty and condensate mass flow for given inlet and geometrical parameters. As mentioned, R134a or similar refrigerant is often used as in heat pump circuits, so the model will take into account change of phase on the tube side. Discretization of differential equation for heat and mass balance and discretization of evaporator geometry – meshing, will be performed, based on selected model complexity, considering computational resources and required precision. Model computation time should be sufficiently low, so that is capable of basic geometric optimization. A secondary target is to develop a semi-automatic procedure that will manage data transfer from the model into the simulation in Starccm+, commercial CFD package, using scripts and macros. Data, that will be transferred from MATLAB simulation, represent heat and mass source terms in CFD code. Agreement between results of the model and CFD simulation will be discussed Developed model and data transfer will be also validated in multiple stages, to test all necessary parts, based on experimental data acquired from test rig built for this application. Agreement of the model to experimental data will be discussed in summary of this thesis. The short version of user manual for the model will be included. As a further simplification of model usage, POD (*Proper Orthogonal Decomposition*) method along with RBF (*Radial Basis functions*) will be implemented, based on previous works and papers from a literature review. An automatic scripted procedure of geometry definition, inlet conditions specification and heat/mass sinks definition in CFD code will be implemented, which will also reduce overall time requirements for the model set-up and the data transfer.

2 Background

Basic research of historical approaches to condensation modeling and standard methods for heat exchanger analysis is performed in order to understand computation methods more deeply, predict its limitations and evaluate suitability for application in this thesis.

2.1 History of condensation modeling

The first theoretical solution was elaborated by Nusselt [2] and it is focused on condensation of pure water vapor on a vertical wall with constant temperature. Nusselt's solution contained many simplifications and assumptions. Later on, Bromley [3] and Roshenow [4] tried to generalize his solution, by taking into account temperature profile in water film and effect surface tension on water/vapor interface. There are only specific industry applications where pure water vapor condensation occurs, more common case is condensation of water vapor with non-condensable gas, usually air. Sparrow and Minkowycz [5] dealt with the problem of forced convection along the horizontal plate. Many experimental researchers also contributed to this topic. Lebedev [6] conducted many experiments that concluded total heat duty increases with Reynolds number and inlet air humidity. Mamyoda and Asano [7] investigated condensation of water vapor with air and managed to create a simple mathematical model for the case of horizontal tubes. A different approach to handle the problem of condensation took Colburn and Hougen [8]. They used the analogy between heat and mass transfer (Sherwood analogy) and managed to create a theory of condensation, that is driven by the concentration gradient of water vapor in the boundary layer of non-condensing gas. They also described total heat duty as a sum of sensible and latent heat. Che and others [9] conducted experiments to verify work of Colburn and Hougen and results shown that measured convective heat transfer coefficient during condensation is 1,5-2x higher than without condensation. Later on, Groff [10] introduced his numerical model for condensation in vertical tubes. His model solved parabolic differential conservation equations without any additional correlations for heat and mass transfer. A modern approach to model heat exchangers, or and heat/mass transfer in general, is numerical simulation using porous medium, that can be used for various types of HE geometry without a need for high mesh resolution, resulting in high cell count. This approach also simply takes into account pressure drop with viscous and porous resistance. Patankar and Spalding [11] were one of the first who used this approach for modeling heat and mass transfer.

2.2 Standard methods for heat exchanger analysis

2.2.1 ε-NTU method

The method of effectiveness (ε-NTU) is widely used computation method, that uses effectiveness ε and *NTU* (Number of Transfer Units) parameters to evaluate heat duty and outlet parameters. First used by London and Seban. [12] Input data for this method are inlet temperatures and mass flows for both flows. The advantage of this method is that outlet temperatures are not necessary for computation, so iterative solution is not required. Constant material properties are assumed in this method.

Effectiveness is defined as the ratio between actual heat duty and maximal heat duty.

$$\varepsilon = \frac{\dot{Q}}{\dot{Q}_{max}} \tag{1}$$

Maximal heat duty is given by equation (2).

$$\dot{Q}_{max} = C_{min}\left(T_{a,in} - T_{c,in}\right) \tag{2}$$

NTU parameter is defined as

$$NTU = \frac{kA}{C_{min}}. \tag{3}$$

Where *A* overall is overall heat exchange area and *k* is overall heat transfer coefficient.The heat capacity ratio is defined as $C_r = \frac{C_{min}}{C_{max}}$. The general relation between ε and NTU for arbitrary heat exchanger geometry and heat capacity ratio is then expressed as

$$\varepsilon = f(NTU, C_r). \tag{4}$$

Outlet temperatures for both flows are evaluated from equations (5) and (6).

$$\dot{Q} = \dot{m}_a \bar{c}_{p,a}\left(T_{a,in} - T_{a,out}\right) \tag{5}$$

$$\dot{Q} = \dot{m}_c \bar{c}_{p,c}\left(T_{c,out} - T_{c,in}\right) \tag{6}$$

Equations for effectiveness ε can be found in literature, tables and other technical publications for given geometry of heat exchanger. As an example, equation (7) describes effectiveness for cross flow plate heat exchanger. [12]

$$\varepsilon = 1 - exp\left[\left(\frac{1}{C_r}\right) NTU^{0.22}\{exp[-C_r NTU^{0.78}] - 1\}\right] \tag{7}$$

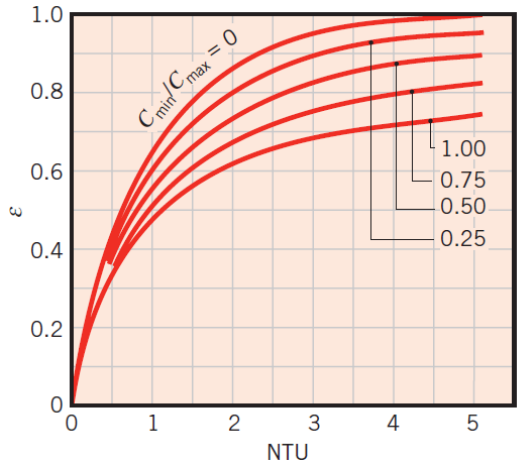


Fig. 2 ε-NTU for the case of crossflow plate [13]

This method can be easily modified for the case with condensation/evaporation simply by putting $C_r \to 0$.

2.2.2 LMTD method

Logarithmic mean temperature difference method (*LMTD*) was first used by Bowman [14]. Heat duty (rate of heat transfer) is expressed as a function of heat transfer are *A*, overall heat transfer coefficient *k* and mean (logarithmic) temperature difference ΔT_{log}. For more complex geometry of heat exchanger or flow arrangements, correction factor *F* is introduced.

($F \leq 1$) In order to evaluate heat duty, outlet temperatures of both flows have to be known, so iterative solution is necessary. Constant flow parameters are also assumed, which may not be negligible in some cases. This problem can be solved by dividing whole heat exchanger into segments and solve in per partes.

Heat duty (rate of heat transfer) is given by equation [15]

$$Q = kAF\Delta T_{log}. \tag{8}$$

Where mean logarithmic temperature difference for co-current heat exchanger is defined as

$$\Delta T_{log} = \frac{\Delta T_{in} - \Delta T_{out}}{\ln\left(\frac{\Delta T_{in}}{\Delta T_{out}}\right)}. \tag{9}$$

For correction (geometrical) factor *F* Bowman, Mueller and Nagle derived an equation based on experimental data and the previous work of Smith and Nusselt. [14]

$$F = \frac{r_{cross}}{r_{counter}} \tag{10}$$

$$r_{counter} = \frac{p - q}{ln\left(\frac{1-q}{1-p}\right)} \tag{11}$$

$$r_{cross} = \sum_{u=0}^{\infty}\sum_{v=0}^{\infty}\left\{(-1)^{u+v}\cdot\frac{(u+v)!}{u!\cdot(u+1)!\cdot v!\cdot(v+1)!}\cdot\left(\frac{p}{r_{counter}}\right)^{u}\cdot\left(\frac{q}{r_{counter}}\right)^{v}\right\} \tag{12}$$

Where p a q are heat transfer effectiveness on the cold (coolant) and the hot (air) side.

$$R = p = \frac{T_{a,in} - T_{a,out}}{T_{a,in} - T_{c,out}} \tag{13}$$

$$P = q = \frac{T_{c,out} - T_{c,in}}{T_{a,in} - T_{c,in}} \tag{14}$$

Equation (12) shows that this method can be quite demanding for heat exchanger with complex geometry, mainly because of the double summation. Tables of graphs can be used instead, but a decrease in precision should be expected.

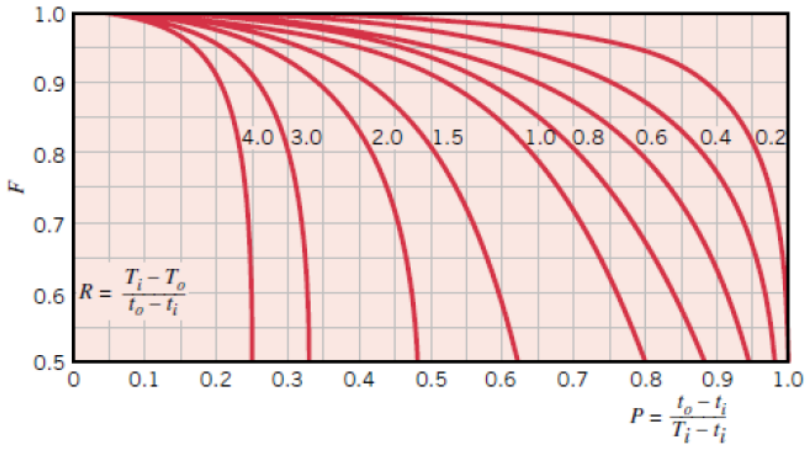


Fig. 3 Graph for factor F as a function of R and P [12]

Other standard methods for heat exchanger analysis can be stated: $P - NTU_t$, that is a variation of $\varepsilon - NTU$ method modified for tube-shell heat exchanger and $\psi - P$ method

proposed by Smith and Mueller, that uses ψ parameter to solve heat duty, defined as $\psi = \frac{\Delta t_m}{T_{a,in} - T_{c,in}}$. All these methods are similar and relation between them describes equation (15).

$$\psi = \frac{\varepsilon}{NTU} = \frac{P}{NTU_t} \tag{15}$$

2.2.3 Ad-hoc numerical methods

Both previously mentioned methods can be used as sub-procedures of a more complex ad-hoc model. As an example of a more complex numerical method is work *Condensation of vapor in the presence of non-condensable gas in condenser* [16] from authors Jen-De Li, Mohammad Saraireh and Grama Thorpe. The target of this work is to develop mathematical model for co-current vertical tube condenser. Humid air and water film flow through the inner tube and cold water through the outer tube. This discrete model provides detailed information about local flow parameters of both media as functions of *x* coordinate (along the tube). Model discretization is shown in Fig. 4.

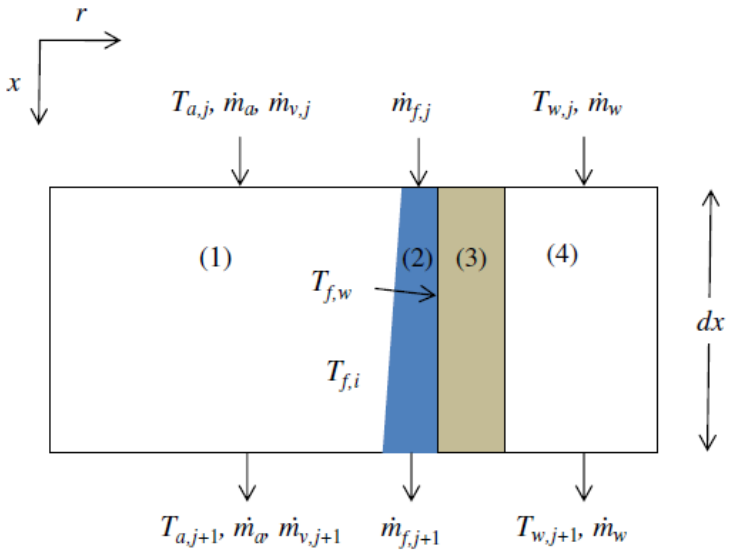


Fig. 4 Model discretization [16]

Authors Ruivo, Dominguez and Costa took a different approach to solution in their work [17], where the modified $\varepsilon - NTU$ method is used. A heat exchanger is divided into multiple sections solved in series, which increases precision, because flow physical properties are changed according to local parameters.

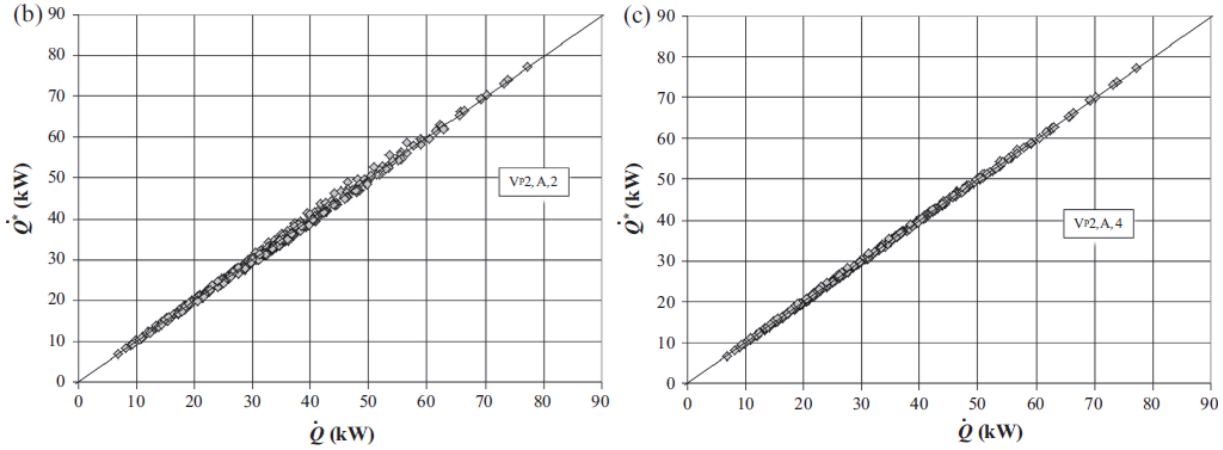


Fig. 5 Comparison of results [17]

As an example of a modern method, the work of Wahabi, Ghorab and Entchev [18] can be stated. They use full 3D CFD analysis for heat transfer modeling inside finned tube heat exchanger. Cold air flows between fins and hot water inside tubes. The target of this work is to study dependency of local flow parameters and overall heat duty on inlet velocity profile and heat exchanger geometry. Results comparison for constant and linear inlet velocity profile is shown in Fig. 6.

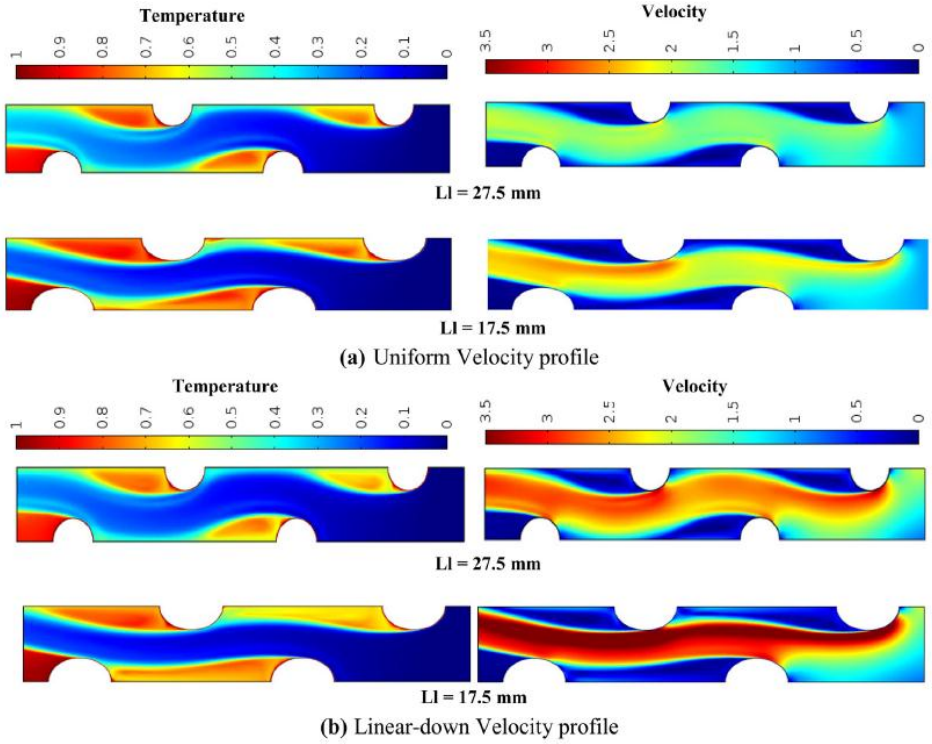


Fig. 6 Comparison of vel. and temp. profile for constant and linear inlet vel. profile [18]

More advanced approach to heat transfer modeling inside heat exchangers is presented in work *POD based modeling on flow and heat transfer of air-cooled condenser influenced by natural wind* [19]. Authors use *POD* (*Proper Orthogonal Decomposition*) method and multiple discrete solutions for selected inlet parameters, to reconstruct arbitrary point from parameter space. This approach allows for a decrease of computation time, despite lower precision of results compared to direct numerical simulation or CFD simulation. An important result from this work number of POD modes that is necessary for a good approximation. From Fig. 7, it is obvious, that using 3-4 modes is sufficient even for more complex geometry.

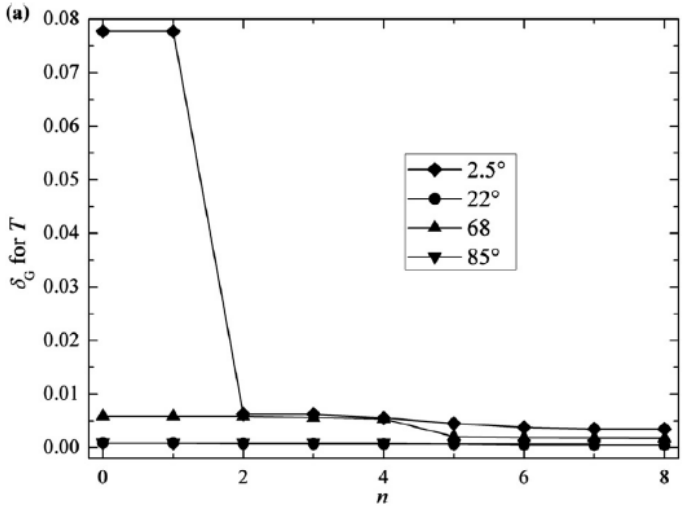


Fig. 7 Dependence of relative error on number of POD modes [19]

Jakub Adamec concluded similar result in his work [20]. Author models heat and mass transfer in fill material in cooling towers. Developed 1D model includes POD method in multiple modifications. Results are compared to direct numerical solution and a similar number of POD modes required to good approximation is concluded. A total number of variable parameters has an effect on the precision of results, reconstructed with POD and RBF methods and it is deeply analyzed in this work. Results for 2 variable parameters and its error are shown in Fig. 8.

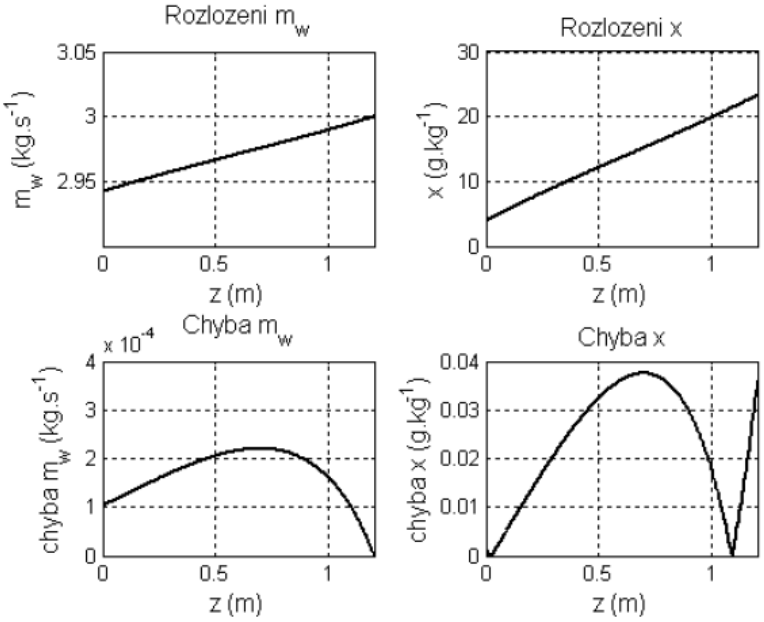


Fig. 8 POD solution and its error compared to direct solution for 2 variable parameters [20]

3 Development of MATLAB model

The mathematical model is developed in MATLAB SW, because of user-friendly environment, simple implementation of matrix operations and ability to use advanced optimization tools for model validation and development. The Model contains simple menu interface for user convenience and tool for data transfer into commercial CFD code. Optimization of code will be performed so model can be used for optimization of heat exchanger geometry.

3.1 Model requirements and properties

The mathematical model simulates heat transfer between humid air and coolant/refrigerant flowing through tubes. Next, simulates mass transfer on outer side - water vapor condensation from air into water film on fins and tubes, and refrigerant evaporation on internal (tube) side. The model predicts locations, where condensation occurs, based on local flow parameters and local fin/tube temperature. Model is parametric in inlet conditions (air/refrigerant temperature and mass flow, humidity, pressure) and also in heat exchanger geometrical parameters (width, height, length, a number of fins, a number of tubes, tube pitch).

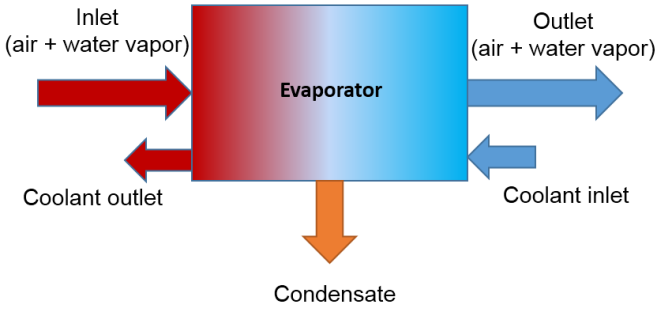


Fig. 9 Evaporator diagram

3.2 Model complexity

A very important attribute of the model is its dimension – in how many axis parameter/property distribution is solved. Choice of model dimension and its complexity affects precision, computation time and computational resources.

- **1D** model is not very suitable for this application, because important parameter for condensation - surface temperature of fin cannot be simplified to 1D case, so big error in condensate mass flow can be expected and thus big error in overall heat duty. Results from this model also would not be very well compatible with more complex CFD simulation
- Full **3D** simulation provides highest precision of outlet parameters (if model is properly calibrated), but model will be very demanding – long computation time, high computation resources requirements. Development of this model is also demanding, but would provide best compatibility with more complex CFD simulation.
- To obtain a balance between model precision, computation time and computation requirements, **2D** model is used. 2 dimensions (coordinate *x* and *y*) where flow parameters are solved correspond to 2 (plate) edges of a fin as shown in Fig. 10.

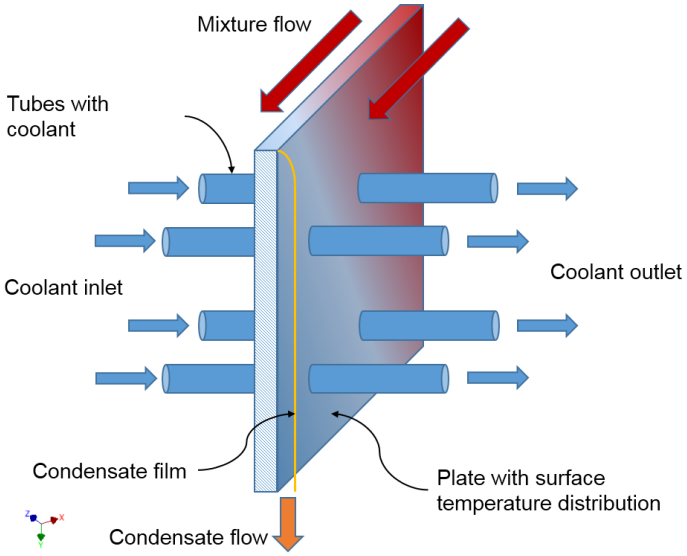


Fig. 10 2D model schema

From point of view of data transfer into commercial CFD, 2D model is not the best, because in remaining *z* coordinate, all parameters are constant. This fact might cause imprecise results if big non-uniformities in z coordinate are present – for example non-uniform velocity in inlet caused by inlet duct.

3.3 Governing equations

Differential equations describing heat and mass transfer problems are listed below. Equations (16) and (17) describe the general balance of mass for air and film flow. Ad hoc equation (18), (19) and (20) describe conservation of enthalpy in 1 slot (space between 2 neighboring fins). Discretization of the equation (21) is used for a surface temperature analysis.

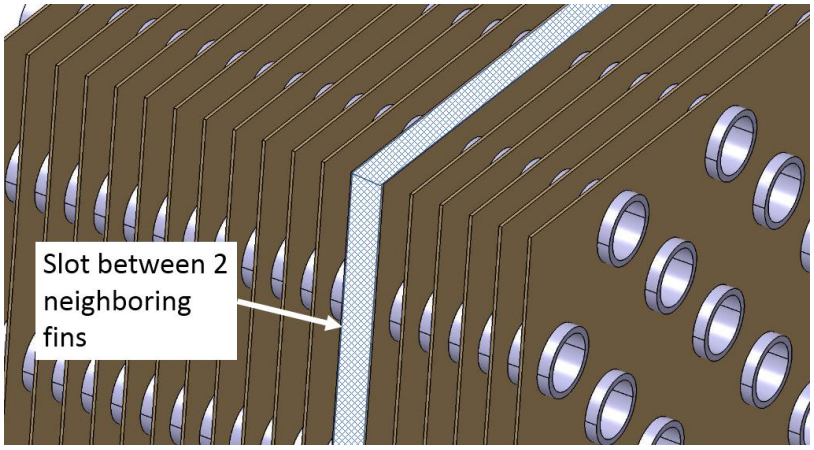


Fig. 11 Flow region - slot

- Continuity equation for air

$$\frac{\partial \rho_a}{\partial t} + \frac{\partial (\rho u_{a,i})}{\partial x_i} = 0; i = \{1,2,3\} = \{x,y,z\} \tag{16}$$

- Continuity equation for water film

$$\frac{\partial \rho_f}{\partial t} + \frac{\partial (\rho_f u_{f,i})}{\partial y} = 0 \tag{17}$$

- Enthalpy conservation equation for humid air

$$dQ_{a\to f} = 2q_{a\to f}dxdy = dH_a - V_a dp_a \tag{18}$$

- Enthalpy conservation equation for water film

$$-dQ_{f\to w} + dQ_{a\to f} = \left(-q_{f\to w} + q_{a\to f}\right)dxdy = dH_f \tag{19}$$

- Enthalpy conservation equation for refrigerant/coolant

$$dQ_{w\to c} = dH_c - V_c dp_c \tag{20}$$

- Fourier's law for heat convection

$$\vec{q} = -\lambda \cdot \boldsymbol{\nabla} T \tag{21}$$

- Equation of state for ideal gas

$$p = \rho RT \tag{22}$$

3.4 Dimensionless numbers

Dimensionless numbers used in heat and mass transfer equations and correlations are listed below. Numbers are evaluated for local flow properties.

- Reynolds number:

$$Re = \frac{uL}{\nu} \tag{23}$$

- Prandtl number:

$$\mathrm{Pr} = \frac{\nu}{a} = \frac{\rho c_p \nu}{\lambda} \tag{24}$$

- Nusselt number:

$$Nu = \frac{\alpha L}{\lambda} \tag{25}$$

- Schmidt number:

$$Sc = \frac{\nu}{D} \tag{26}$$

- Sherwood number:

$$Sh = \frac{\beta L}{D} \tag{27}$$

- Lewis number:

$$Le = \frac{a}{D} = \frac{Sc}{Pr} \tag{28}$$

- Lewis factor:

$$Le_f = \frac{\alpha}{c_p \beta} \tag{29}$$

3.5 Simplifying assumptions

Modeling heat and mass transfer in complicated geometry is a complex problem, so couple simplifications are assumed: [21]

1) The fin is flat. (corrugated fins are usually present, the effect of corrugation in included in *AreaFx*, *AreaFy* parameters – area increase factors – see *note 1*)
2) For the first guess, refrigerant temperature is constant (equal to boiling temperature) through whole evaporator. During simulation, refrigerant temperature distribution is developing.
3) Condensate from tubes is dropping down, not flowing down the fin.
4) Temperature of fin is constant through its thickness.
5) No heat flow from fin edges.
6) All fields of flow parameters (temperature, velocity, humidity …) are constant in z direction in every slit.
7) Steady-state flow.
8) Perfect contact between outer tube surface and fin.

Note 1.:Area increase factors are simple means of taking into account waves on fins in both directions and its effect of increasing total heat transfer area of the fin. However, this approach does not take into account aerodynamical effect caused by these waves.

Parameter *AreaFy* considers fin area increase due to "waves" in *y* direction: (*AreaFy > 1, AreaFx = 1*)

Fig. 12 Schema of waves in y direction

Similarly, parameter *AreaFx* considers fin area increase due to "waves" in *x* direction: (*AreaFx > 1, AreaFy = 1*)

3.6 Model discretization and equations for heat and mass balance

Finding a continuous solution for heat exchangers with complex geometry and general flow arrangement is very complicated, almost impossible. So, searching for a discrete numerical solution is the only option (except measurement of parameters on test rig), but also sufficient option in most cases. A discrete solution requires mesh/element to be defined, in this case, structured mesh with rectangular elements is used. Algorithms for generating tube circular cross-section in a rectangular grid are developed.

3.6.1 Coordinate system and flow orientation

Heat exchanger with arbitrary dimensions and its coordinate system is in Fig. 13. (x – length, y – height, z - width)

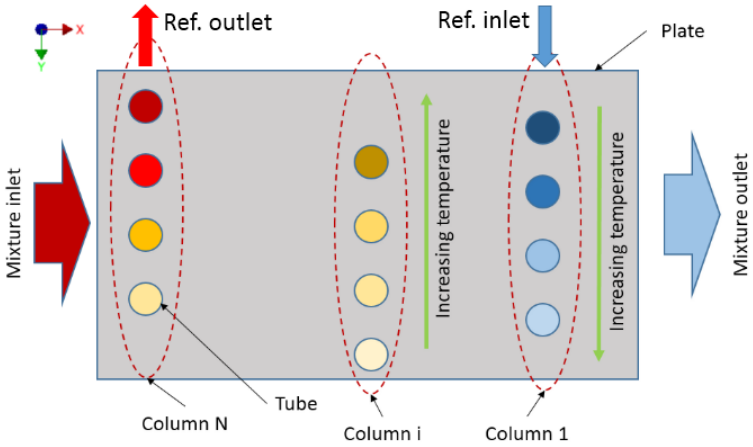


Fig. 13 Coordinate system orientation

Air flow is always orientated in x direction (from left to right), and refrigerant flow is oriented in z direction – "inlet tube section" is placed at upper right corner of fin and "outlet tube section" at upper left corner (only even number of tube columns with staggered pattern is supported in this version).

3.6.2 Meshing

Automatic mesh generation (and tube bundle generation) on fin is included in the model. (only parameter of the mesh is *Telem* - #of elements per tube cross-section). Number of elements in *x* direction - *N* and y direction - *M* is solved automatically for given *Telem* parameter by sub-optimization (minimization) of tube cross section are error - $ErrSurf$.

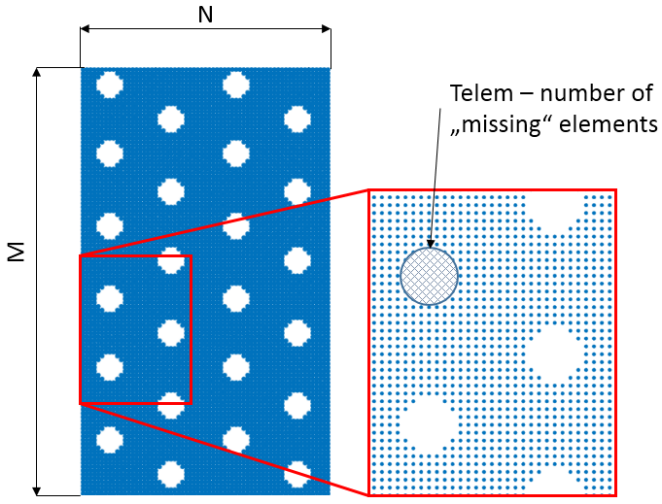


Fig. 14 Automatic mesh

For numerical simulation, it is necessary to divide tube into elements/segments in which temperature/enthalpy increments are evaluated. In this model, tube element is defined by tube segment between 2 neighboring fins.

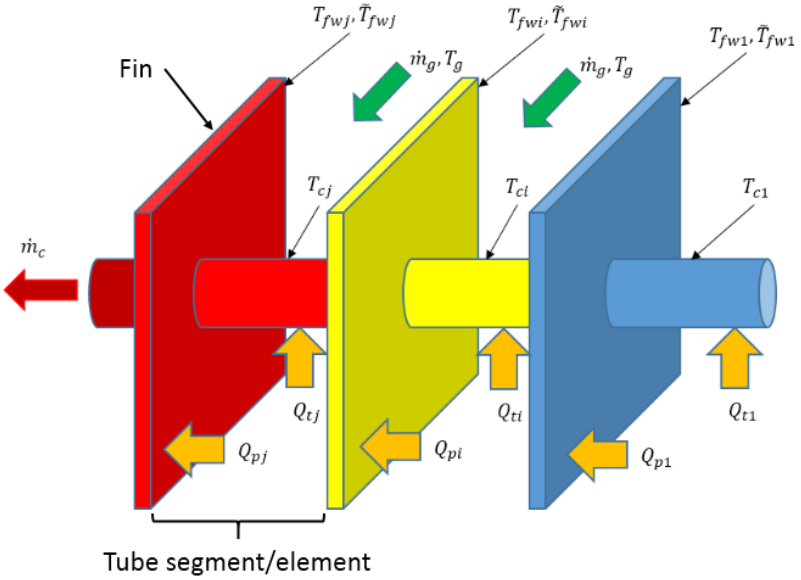


Fig. 15 Tube segment/elements definition

3.7 Discrete equations for heat and mass balance

Differential equations in incremental form for heat and mass balance in 1 slot (space between 2 neighboring fins), between air, film and fin/tube surface, used to solve local air temperature, humidity, film temperature and mass flow. [22] [21]

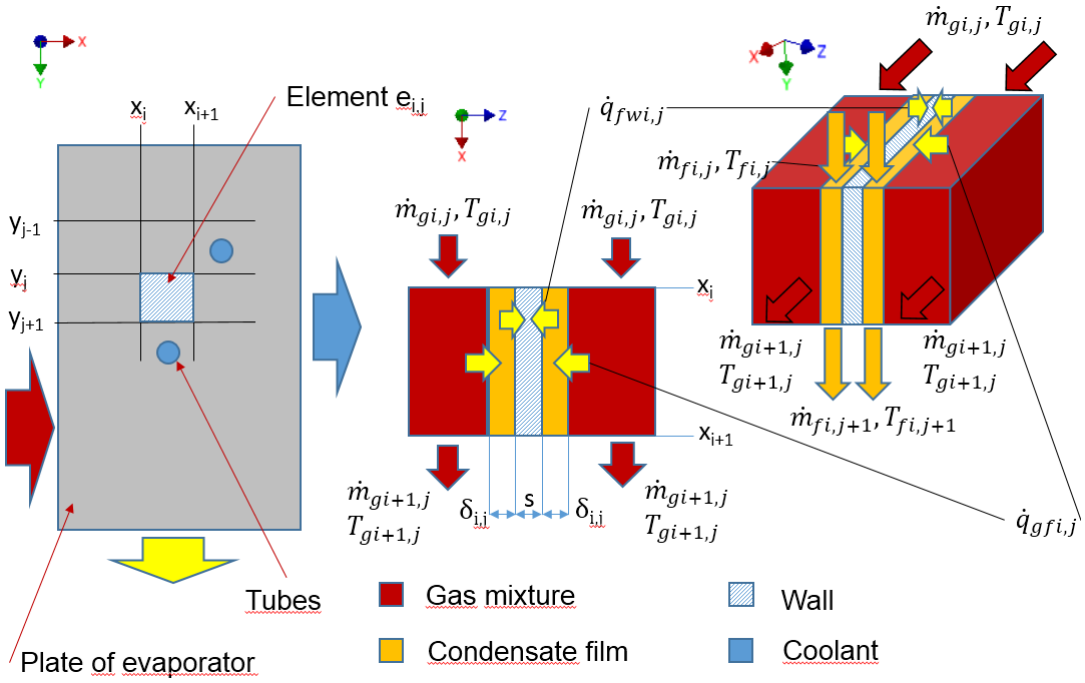


Fig. 16 Discretization

Air/film heat balance:

$$\dot{m}_{ai,j}c_{p,a}T_{gi,j} + \dot{m}_{vi,j}\left(l_{230} + c_{p,v}T_{gi,j}\right) = \dot{m}_{ai+1,j}c_{p,a}T_{gi+1,j} + \dot{m}_{vi+1,j}\left(l_{230} + c_{p,v}T_{gi+1,j}\right) + \alpha_{gf}E_T\left(T_{gi,j} - \bar{T}_{fi,j}\right)dxdy \quad (30)$$

Film/air/surface heat balance:

$$\dot{m}_{fi,j}c_f\bar{T}_{fi,j} = \dot{m}_{fi,j+1}c_f\bar{T}_{fi,j+1} - \alpha_{gf}E_T\left(T_{gi,j} - \bar{T}_{fi,j}\right)dxdy + \alpha_{fw}\left(\bar{T}_{fi,j} - T_{fwi,j}\right)dxdy \quad (31)$$

Refrigerant mass balance:

$$\dot{m}_{c,k} = \dot{m}_{c,k+1} \tag{32}$$

Overall mixture (air) and film mass balance:

$$\dot{m}_{fi,j} + \dot{m}_{gi,j} = \dot{m}_{fi,j+1} + \dot{m}_{gi+1,j} \tag{33}$$

Film mass balance:

$$\dot{m}_{fi,j} + \Delta\dot{m}_{fi,j} = \dot{m}_{fi,j+1} \tag{34}$$

Mixture (air) mass balance:

$$\dot{m}_{gi,j} = \dot{m}_{gi+1,j} + \Delta\dot{m}_{fi,j} \tag{35}$$

Additional equations:

$$\dot{m}_{gi,j} = \dot{m}_{ai,j} + \dot{m}_{vi,j}; \tag{36}$$

$$\dot{m}_{gi+1,j} = \dot{m}_{ai,j} + \dot{m}_{vi+1,j}; \tag{37}$$

$$\left|\Delta\dot{m}_{vi,j}\right| = \left|\Delta\dot{m}_{fi,j}\right| \tag{38}$$

Equations above form a system of differential equations in incremental form. Explicit first order Euler method is used to solve this system.

Local heat and mass transfer coefficients are obtained from correlations for the case of flow in small slot (as a limit of concentric annular duct) and flow over staggered tube bundle. These coefficients are then used to solve overall heat transfer coefficient according to following picture. [22] [21]

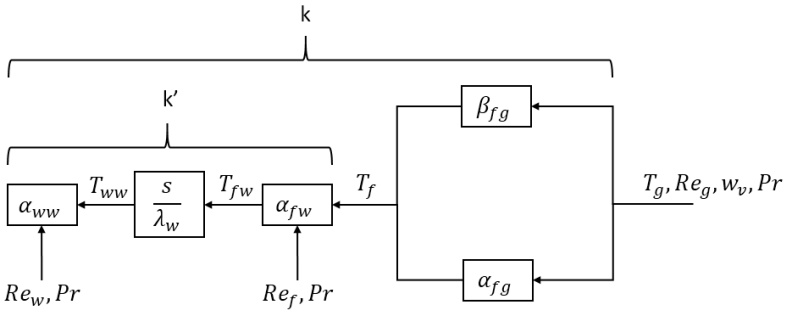


Fig. 17 Overall HTC

3.8 MATLAB solver sub-procedures

MATLAB model itself is composed of multiple sub-procedures, which solve various heat and mass transfer problems. [21]

- Surface temperature solver
- Refrigerant temperature solver
- "i + j" cycles to solve air and film local parameters (Air/film enthalpy solver)
- Film thickness solver
- Film temperature solver

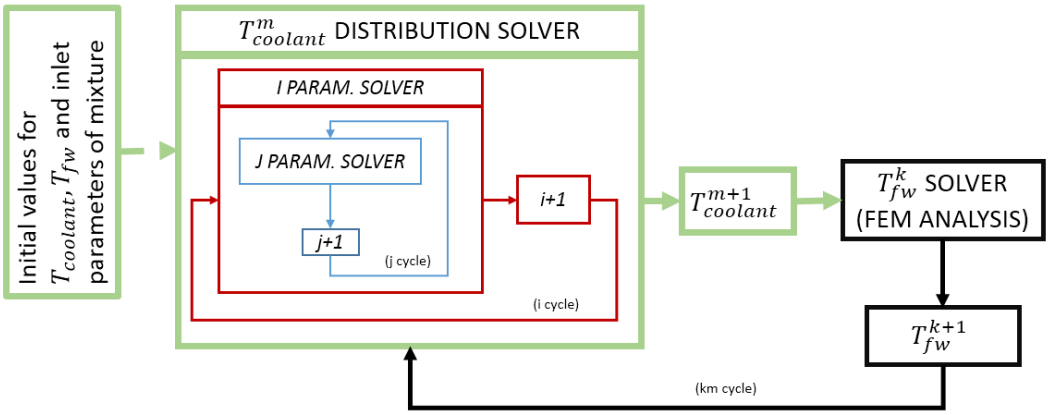


Fig. 18 MATLAB model - solver diagram

The iterative solution procedure is obvious from the diagram above. First, the initial distribution of air temperature and condensate flow field is solved from guessed refrigerant temperature distribution. Based on new air temperature, a new refrigerant temperature distribution is solved, from which new fin surface temperature is evaluated. New refrigerant and surface temperature are then used to solve for new air temperature and condensate flow and iteration loop is completed. A number of iterations required to obtain stable solution depends on inlet conditions, but varies from 50 to 150.

Under-relaxation factors are included in *Surface temperature solver* and *Refrigerant temperature solver*, to avoid solution instability during first couple iterations.

3.8.1 Film thickness solver (viscosity effect in condensate film)

Because the focus is on evaporators for air dehumidification, low condensate flow is expected, thus, small thickness of a film is assumed. Velocity gradient for a simple 1D case of fluid film on vertical wall can be then derived as [16]

$$\frac{du_f}{dz} = \frac{\rho_f - \rho_g}{\mu_l} g(\delta - z) + \frac{\tau_g}{\mu_l}. \tag{39}$$

Ordinary differential equation above is integrated and boundary conditions applied, velocity profile is obtained. [16]

$$u_f(z) = \frac{\rho_f - \rho_g}{\mu_l} g \left[\delta z - \frac{1}{2} z^2\right] + \frac{\tau_g}{\mu_l} z \tag{40}$$

Where range for z coordinate is $0 \leq z \leq \delta_{j,i}$, ρ_f – condensate film density, ρ_g – mixture density and τ_g is shear stress at interface between mixture and film.

Local mean velocity in condensate film is obtained by integration over local film thickness

$$\bar{u}_f(x,y) = \bar{u}_{f,j,i} = \frac{1}{\delta_{j,i}} \int_0^{\delta_{j,i}} u_f(x,y,z)\, dz \tag{41}$$

Local mass flow of condensate film:

$$\dot{m}_{f,j,i} = \rho_f \bar{u}_{f,j,i} \delta_{j,i} dx \tag{42}$$

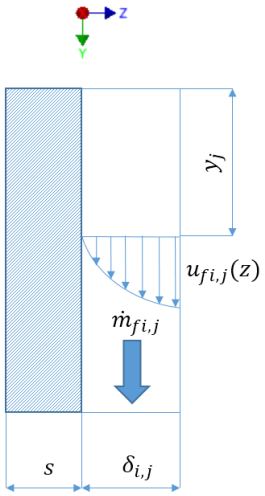


Fig. 19 Viscosity effect in condensate film

These equations are used to solve local film thickness - $\delta_{i,j}$, that is necessary for solving heat transfer through film.

3.8.2 Film temperature solver

Local temperature of film on tubes and fins is very important parameter for heat transfer and condensation and it is determined from heat balance between humid air, film and fin/tube surface [23].

Heat balance between air, film and surface can be written as

$$k'(T_F - T_{fw}) = \dot{m}_f{}''\Delta h_{23} + \alpha_{fg} E_T (T_G - T_F). \tag{43}$$

Increment of film mass flow:

$$\dot{m}_f{}'' = \rho_g \beta_g ln\left(\frac{1-\widetilde{y_v}\{T_F\}}{1-y_v}\right) \tag{44}$$

$$\phi_T = \frac{\rho_g \beta_g c_{pv}}{\alpha_{fg}} ln\left(\frac{1-\widetilde{y_v}\{T_F\}}{1-y_v}\right) \tag{45}$$

Combining equations above we obtain:

$$k'(T_F - T_{fw}) = \alpha_{fg}\phi_T\left(\frac{\Delta h_{23}}{c_{pv}} + \frac{T_G - T_F}{1 - exp(-\phi_T)}\right) \tag{46}$$

Equation (46) is solved iteratively:

$$\rightarrow T_F^{i+1} = T_c + \frac{\alpha_{fg}\phi_T}{k'}\left(\frac{\Delta h_{23}}{c_{pv}} + \frac{T_G - T_F^i}{1 - exp(-\phi_T)}\right) \tag{47}$$

From equation (47), mean film temperature is solved after couple iterations. This equation is also used as a part of solution stability monitoring – the first indication of solution divergence is unstable film temperature, if this fact is detected, simulation is terminated.

3.8.3 Superposition of heat and mass transfer

Theoretical heat transfer coefficient on fin and tube is known, but effect of tube presence on heat transfer coefficient on fins (and fin presence to heat transfer coefficient on

tubes), needs to be included. If we assume that evaporator is only composed of tubes, total heat duty without condensation (mass transfer) can be written symbolically as: [21]

$$Q_{tubes} = \tilde{\alpha}_{tube} \cdot A \cdot \overline{(T_g - T_f)}, \tag{48}$$

where $\tilde{\alpha}_{\mathrm{tube}}$ is heat transfer coefficient on tubes. Similarly, if evaporator (heat exchanger) is made only from fins, total heat duty is:

$$Q_{fins} = \tilde{\alpha}_{plate} \cdot A \cdot \overline{(T_g - T_f)}, \tag{49}$$

where $\tilde{\alpha}_{plate}$ is heat transfer coefficient on fins/plates.

Superposing fins and tubes, total heat duty without condensation (mass transfer) is symbolically:

$$Q_{tot} = \left(R_p \cdot \tilde{\alpha}_{plate} + R_t \cdot \tilde{\alpha}_{tube}\right) \cdot A \cdot \overline{(T_g - T_f)}. \tag{50}$$

Where superposition (effectiveness) coefficients R_p, R_t are derived from experimental data by optimization – fitting the model results to results from experiment. These coefficients describe influence of temperature and velocity field of one case to the other. Similar hypothesis can be made with condensate mass flow, so another 2 effectiveness coefficients are introduced.

3.8.4 Plate/fin surface temperature solver

After considering simplifications (mainly points 1,4 and 5), problem of fin surface temperature can be then solved separately (based on informations from air and refrigerant side) as 2D heat conduction problem in rectangular domain. To ensure unique solution, boundary conditions are required – refrigerant temperatures at fin/tube intersections are used. As a volumetric heat loads, local heat flow from gas (or film) into fin is used. [24] [21]

Necessary equations for discrete solution of heat conduction in 2D domain using FEM.

Heat flow (flux) vector

$$\boldsymbol{q} = \begin{bmatrix} q_x \\ q_y \end{bmatrix}. \tag{51}$$

Fourier's law in matrix form

$$\boldsymbol{q} = -\boldsymbol{D}\boldsymbol{\nabla}T. \tag{52}$$

Linear shape functions (***N***) are chosen for interpolation, weak formulation is established as

$$\iint_{\Omega} \boldsymbol{B}^T s \boldsymbol{D} \boldsymbol{B} d\Omega \boldsymbol{a} = -\oint_{\partial\Omega_q} \boldsymbol{N}^T t q_b d\partial\Omega - \oint_{\partial\Omega_T} \boldsymbol{N}^T s q_n d\partial\Omega + \iint_{\Omega} \boldsymbol{N} s Q d\Omega. \tag{53}$$

Where

$$\boldsymbol{B} = \nabla \boldsymbol{N}, \tag{54}$$

$$\boldsymbol{a} = [T_i]. \tag{55}$$

Like in elasticity, stiffness matrix, boundary terms and load vector are defined as:

The stiffness matrix

$$\boldsymbol{K} = \iint_{\Omega} \boldsymbol{B}^T s \boldsymbol{D} \boldsymbol{B} d\Omega. \tag{56}$$

Boundary terms (boundary load)

$$\boldsymbol{f}_b = -\oint_{\partial\Omega_q} \boldsymbol{N}^T t q_b d\partial\Omega - \oint_{\partial\Omega_T} \boldsymbol{N}^T t q_n d\partial\Omega. \tag{57}$$

Internal load vector (volumetric heat sources):

$$\boldsymbol{f}_l = \oiint_{\Omega} \boldsymbol{N} t Q d\Omega. \tag{58}$$

Equations from weak formulation can be then rewritten as a simple matrix equation:

$$\boldsymbol{K}\boldsymbol{a} = \boldsymbol{f}_b + \boldsymbol{f}_l = \boldsymbol{f}. \tag{59}$$

This method is tested on following example:

At the corners and in the middle of a square (1mm thickness, 100x100 mm) plate different heat of sources are placed. 4 Blue dots represent tubes with a given temperature (8℃). Mesh grid 100x100 is used. Resulting temperature for the example is shown in Fig. 20.

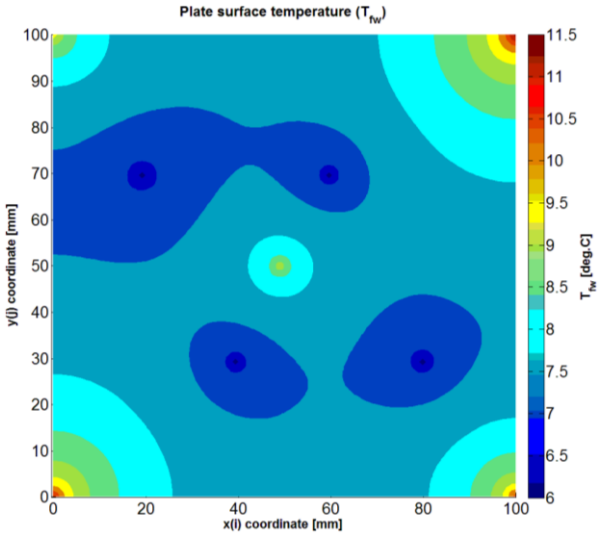


Fig. 20 Testing example for FEM

This method is then applied on heat exchanger geometry for arbitrary inlet conditions, resulting surface temperature is shown in Fig. 21.

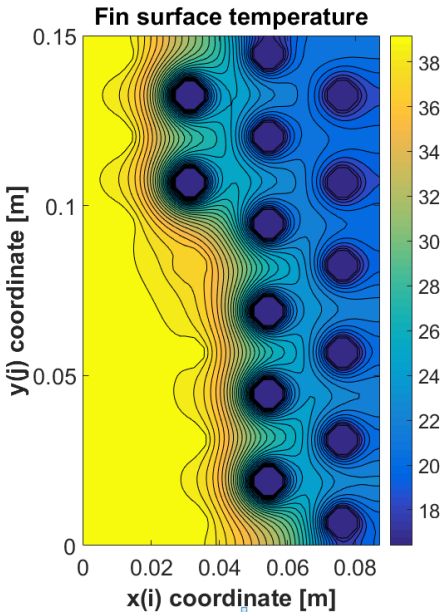


Fig. 21 FEM result during simulation of real heat exchanger

Because plate/fin surface temperature is solved every iteration, code optimization is required to decrease overall computation time even for fine meshes. Temperature in mesh nodes can solver from equation (60) by

$$\boldsymbol{a} = \boldsymbol{K}^{-1}\boldsymbol{f} \tag{60}$$

3.8.4.1 Solver improvements

Matrix inverse operation (equation (60)) is very demanding and imprecise for large matrices. The dimension of stiffness matrix for a coarse mesh is typically 7300x7300, which takes 22 seconds on average (4 core CPU, 3,3GHz). This is not suitable for simulations that usually take 150 iterations to converge. Matrix inverse can be replaced by *mldivide* operation implemented in MATLAB.

$$\boldsymbol{a} = \boldsymbol{K}\backslash\boldsymbol{f} \tag{61}$$

This operation noticeably improves code performance, the computation time of solving equation (61) is decreased to 9 seconds for same coarse mesh, but further improvements are required. Because stiffness matrix $\boldsymbol{K}$ is constant (under an assumption that fin/plate material conductivity is not rapidly changing with temperature), it can be assembled only once, before the iteration procedure starts. Next, typical stiffness matrix for any finite elements problem in 2D is a sparse matrix, so optimized algorithms for this type of matrices is used – sparse matrix definition in triplet form (*CRS* – compressed row storage). This decreases demands on memory and also speeds up stiffness matrix assembly. Combining *mldivide()* and sparse matrices, the computation time of solving equation (62) is decreased to 0.9 seconds.

$$\boldsymbol{a} = \boldsymbol{K}_{sparse}\backslash\boldsymbol{f} \tag{62}$$

At this point, computation time is sufficiently low, but further improvement can be made. The reason for that is the algorithm used by *mldivide()* function, it uses LU factorization modified for sparse matrices. This is inefficient, because *mldivide()* algorithm performs LU decomposition of $\boldsymbol{K}_{sparse}$ every iteration, so it is appropriate to perform decomposition before iteration procedure starts, by using MATLAB function *lu()*.

$$[\boldsymbol{L}_{\boldsymbol{K}}, \boldsymbol{U}_{\boldsymbol{K}}] = lu(\boldsymbol{K}_{sparse}) \tag{63}$$

Equation (62) is then replaced by 2 equations

$$\tilde{\boldsymbol{a}}_{temp} = \boldsymbol{L}_{\boldsymbol{K}}\backslash\boldsymbol{f} \tag{64}$$

$$\boldsymbol{a} = \boldsymbol{U}_{\boldsymbol{K}}\backslash\tilde{\boldsymbol{a}}_{temp}. \tag{65}$$

A solution of 2 equation above takes 0.02 second on average, but additional time for LU decomposition before the simulation is required.

Plate/fin surface also needs to be divided into sub-regions (areas), from where all the heat flow coming from the air is assigned to specific tube cross section. This allows evaluating enthalpy/temperature increment of tube pass. In general, these sub-regions can be traced by finding locations, where local heat flux inside the fin is 0 or very small. A boundary of these sub-regions are very complicated (Fig. 22); so an additional simplification is required.

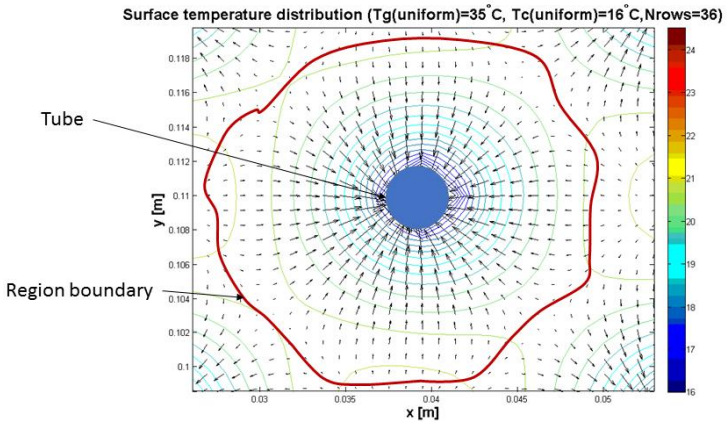


Fig. 22 Complex fin sub-region (arrows - heat flux, colorbar - temperature)

Simplified sub-regions of fin surface are shown in Fig. 23.

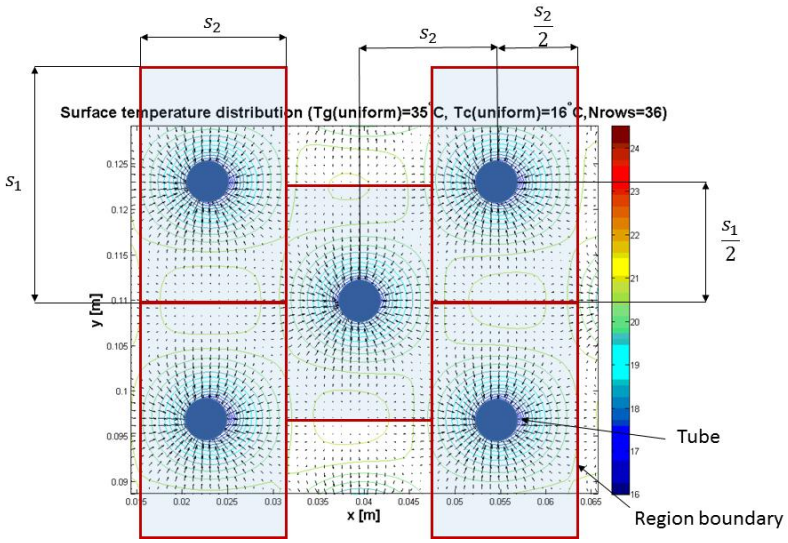


Fig. 23 Simplified fin sub-regions

This simplification also brings new problem – for a case with big temperature gradients between 2 neighboring tubes, some of the heat that in reality "flows" into tube 1 is within this simplification associated with tube 2. This causes an error in heat balance (difference of enthalpy on air side is not equal to enthalpy on the refrigerant side). To prevent this situation, *Redistribution Correction Solver* is added. RCS eliminates this error by transferring additional heat from 1 sub-region to the other, based on estimation from heat flux integral on its boundary.

3.9 Heat transfer coefficients

Essential part of mathematical model are coefficients for heat and mass transfer. Theoretical HT/MT coefficients can be found in many papers, but calibration based on experimental data is necessary to achieve high precision of model outputs.

3.9.1 Correlations for heat transfer coefficients

There are 3 surfaces where HT occurs – outer and inner tube surface and fins. Respective theoretical HT coefficients used in model are described below.

3.9.1.1 Heat transfer coefficient on tube

To model heat transfer between mixture (air and water vapor) and tubes (or film on tubes), well known correlation for staggered tube bundle is used (Fig. 24). Correction for lower number of tube rows is also included. [23]

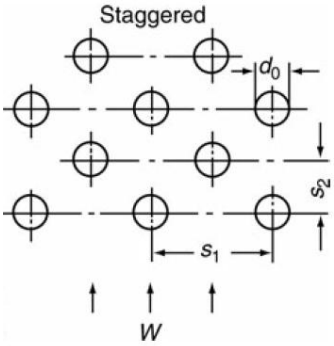


Fig. 24 Tube bundle schema [23]

Characteristic length:

$$l = \left(\frac{\pi}{2}\right) d_0 \tag{66}$$

Pitch ratios:

$$\textit{Transverse} \quad a = \frac{s_1}{d_0}, \tag{67}$$

$$\textit{Longitudinal} \quad b = \frac{s_2}{d_0}. \tag{68}$$

Void fraction:

$$\psi = 1 - \frac{\pi}{4a} \tag{69}$$

Arrangement factor:

$$f_{A,stag} = 1 + \frac{2}{3b}. \tag{70}$$

Nusselt number:

$$Nu_{l,lam} = 0{,}664\sqrt{Re_\psi}\sqrt[3]{Pr} \tag{71}$$

$$Nu_{l,turb} = \frac{0{,}037 Re_\psi^{0,8} Pr}{1 + 2{,}443 Re_\psi^{-0,1}(\mathrm{P}r^{2/3} - 1)} \tag{72}$$

$$Nu_{l,0} = 0{,}3 + \sqrt{Nu_{l,lam}^2 + Nu_{l,turb}^2} \tag{73}$$

$$Nu_{0,boundle} = f_{A,stag} Nu_{l,0} \tag{74}$$

Considering effect of the number of rows by including correction factor

$$Nu_{0,boundle} = \frac{1 + (n-1) f_{A,stag}}{n} Nu_{l,0}. \tag{75}$$

Heat transfer coefficient

$$HTC_t = \frac{Nu_{0,boundle} \cdot \lambda}{l}. \tag{76}$$

Range of validity:

$$Re_\psi = \frac{wl}{\psi\nu} \quad 10 < Re_\psi < 10^6$$

$$Pr = \frac{\nu}{a} \quad 0{,}6 < Pr < 10^3$$

3.9.1.2 Heat transfer coefficient on fin

Heat transfer between mixture (air and water vapor) is modeled with correlations for laminar flow (verified for reasonable range of inlet parameters) in parallel duct, as a limit of correlations for concentric annular ducts, introduced by Martin, Stephan, Mercer et al. [23] Local flow parameters are used to evaluated heat transfer coefficient. It is also assumed, that corrugation on fins does not cause turbulent flow.

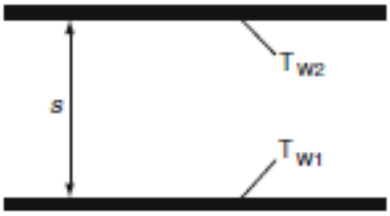


Fig. 25 Rectangular duct [23]

Characteristic length:

$$d_h = 2 \cdot s \tag{77}$$

Nusselt numbers: [23]

Asymptotic Nusselts number for hydrodynamically developed flow

$$Nu_1 = 7.541. \tag{78}$$

Nusselt number for thermally developing flow

$$Nu_2 = 1.841 \cdot \sqrt[3]{\frac{Re \cdot Pr \cdot d_h}{l}}. \tag{79}$$

Nusselt number for hydrodynamically and thermally developing flow

$$Nu_3 = \left(\frac{2}{1 + 22 \cdot Pr}\right)^{1/6} (Re \cdot Pr \cdot d_h/l)^{1/2}. \tag{80}$$

Overall Nusselt number:

$$Nu_m = (Nu_1^3 + Nu_2^3 + Nu_3^3)^{1/3} \tag{81}$$

Heat transfer coefficient:

$$HTC_p = \alpha_{fw} = \frac{Nu_m \cdot \lambda_g}{d_h} \tag{82}$$

Range of validity:

$$0.6 \leq Pr \leq 1000$$

$$0 \leq \frac{d_h}{l} \leq 1$$

$$1 \leq Re \leq 2300$$

3.9.1.3 Heat transfer coefficient on refrigerant side

Well known correlations for heat transfer coefficient inside pipe during turbulent flow and piecewise constant wall temperature (constant outer tube surface temperature is assumed) is used. [23] If local flow parameters of refrigerant satisfy boiling conditions, constant value for HTC is used - $\alpha_{ww,boil} = 10^5\ W \cdot m^{-2} \cdot K^{-1}$. This simplification does not lead to significant error, because main (>96%) heat resistance is located on air side.

Nusselt number:

$$Nu_r = \frac{\xi/8 \cdot Re \cdot Pr}{1 + 12.7 \cdot \sqrt{\xi/8}(\mathrm{Pr}^{2/3} - 1)}\left[1 + \left(\frac{d_i}{l}\right)^{2/3}\right] \tag{83}$$

Friction factor:

$$\xi = (1.8 \cdot \log_{10} Re - 1.5)^{-2} \tag{84}$$

Heat transfer coefficient:

$$HTC_r = \alpha_{ww} = \frac{Nu_r \cdot \lambda_r}{d_i} \tag{85}$$

Range of validity:

$$0.1 \leq Pr \leq 1000$$

$$\frac{d_i}{l} \leq 1$$

$$10^4 \leq Re \leq 10^6$$

3.9.2 Mass transfer coefficients

Modified Sherwood analogy based on work of Kloppers and Kröger [25] is used for mass transfer coefficient evaluation. Method takes into account effect of Lewis factor change based on flow parameters, as described by Bourillot [26] and others, which disapproves assumption of Merkel and Lewis [27], that $Le_f = 1$.

Lewis number for humid air:

$$Le = \frac{Sc_g}{Pr_g} \tag{86}$$

Humidity/saturation parameter:

$$\xi = \frac{x_{SAT} + d}{x + d} \tag{87}$$

Where $d = \frac{Molecular\ weight\ of\ water}{Molecular\ weight\ of\ air} = 0.622$ [25]

Correction function

$$F(\xi) = \frac{\log \xi}{\xi - 1} \tag{88}$$

Lewis factor in Bosnjakovic form

$$Le_f = \frac{Le^{2/3}}{F(\xi)} \tag{89}$$

Mass transfer coefficient

$$MTC = \beta = S_{p,t} \frac{HTC}{c_{p,g} \cdot Le_f} \tag{90}$$

Coefficients $S_p$and S_t are derived from experimental data by optimization (fitting model to experimental data).

3.10 Model validation

Developed mathematical model is validated and calibrated with experimental data, in order to deliver precise and reliable results in wide range of inlet and geometrical parameters. Validation itself is divided into 4 stages, in which all necessary parts of the model are verified. The complex test rig is constructed for reliable data acquisition. Ing. Lukáš Dvořák, Ph.D., Ing. Zdeněk Sumara, Bc. Michal Šochman and Ing. Michal Schmirler, Ph.D. contributed to building this device, development of control software and conduction of experiments.

Fig. 26 Test rig with heat pump circuit

Multiple validations of test rig are performed, to test and verify uniform inlet conditions into a evaporator, humidification, inlet temperature control, total mass flow control, the precision of measured values behind evaporator and condenser etc.

Fig. 27 Test rig construction

3.10.1 Parameter of selected heat exchanger

Validation of mathematical model requires real heat exchanger/evaporator that can be used for measurements on the test rig to obtain necessary data. Dimensions of selected heat

exchanger are listed in Table 1. It is a part of heat pump unit, other components of the circuit are also incorporated into test rig to assure proper refrigerant circulation.

Width (mm)	Height (mm)	Depth (mm)	Number of fins (1)	Fin pitch (mm)	Fin thickness (mm)
234	150	87	117	2	0.12

Table 1 Heat exchanger parameters

Parameters of tube bundle are listed in Table 2.

Tube pattern	Tube outer diameter (mm)	Tube thickness (mm)	Number of rows (1)	Number of tubes per row (1)	Tube x pitch (mm)	Tube y pitch (mm)
staggered	9.52	0.7	12	2	21.65	25

Table 2 Tube bundle parameters

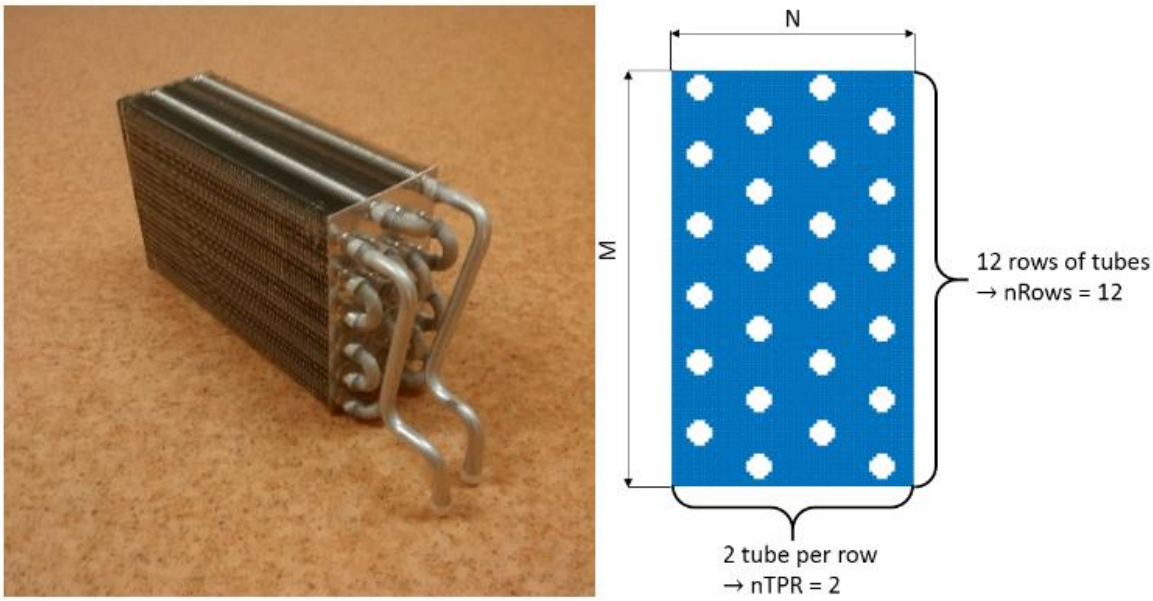


Fig. 28 Real evaporator used to measurements and mesh in profile

Different meshes are applied, depending on application.

- Coarse mesh for fast simulation – $M = 112, N = 65, Telem = 37, ErrSurf = 0.68\%$

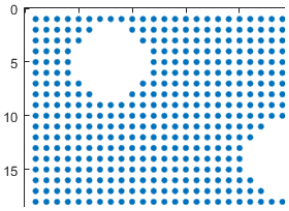


Fig. 29 Coarse mesh section

- Medium mesh for optimization – $M = 174, N = 101, Telem = 97, ErrSurf = 0.71\%$

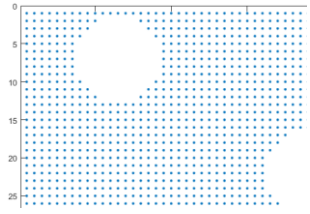


Fig. 30 Medium mesh section

- Fine mesh for data transfer – $M = 238, N = 138, Telem = 185, ErrSurf = 0.33\%$

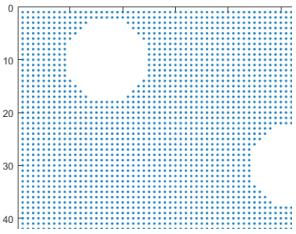


Fig. 31 Fine mesh section

Validation stages:

1. No condensation on air side and water as a coolant
2. Condensation on air side and R134a as a refrigerant
3. Validation of data transfer between MATLAB model and CFD SW

Minor modifications of test rig were required for every stage of validation – changing coolant from water to R134a, creating baffles in from of evaporator, etc. Verification of uniform inlet conditions to evaporator inside test rig is also performed, to eliminate as many variables as possible and provide reliable input to model validation.

3.10.2 Stage 1 validation

The first stage of validation is done with no condensation and water as a coolant. It is focused correction of heat transfer coefficient on the air side. A total of 25 points (Table 3) are measured on the test rig with selected heat exchanger (evaporator), but only 20 points are used for validation, because 5 points excluded due to high (>5%) heat balance error evaluated from experimental data. The main target of this validation is to identify 2 important superposition (effectiveness) coefficients R_p, R_t. Using simplex method (Nelder-Mead simplex algorithm) implemented in *MATLAB – Optimization Toolbox* function *fminsearch*. Target function (fitness) used for optimization procedure is defined as

$$TF_1 = \sum_{i=1}^{n} \left[\begin{matrix} w_g \cdot \left\| T_{g,out,meas,i} - T_{g,out,model,i} \right\| + w_c \cdot \left\| T_{c,out,meas,i} - T_{c,out,model,i} \right\| \\ + w_w \cdot \left\| w_{v,out,meas,i} - w_{v,out,model,i} \right\| \end{matrix} \right]. \quad (91)$$

This functions represents sum of relative errors for all output parameters (air and coolant temperature, outlet humidity) and it is dependent on effectiveness coefficients R_p, R_t. By using MATLAB function *fminsearch()*, a minimum of the target function can be found. This minimum corresponds to best agreement between model and experimental results.

Obtained optimal parameters are identified as

$$R_p = 0.87, \quad (92)$$

$$R_t = 0.98, \quad (93)$$

by using weights $w_g = 1; w_c = 1.5, w_w = 0$. Because only local optimization method is used (multiple runs with different initial guesses to verify the stability of archived parameters), additional improvements are possible using a global method, for example, *genetic algorithms*. Results from validation are shown in Fig. 32 – relative error of outlet air and refrigerant temperature for all points measurement points.

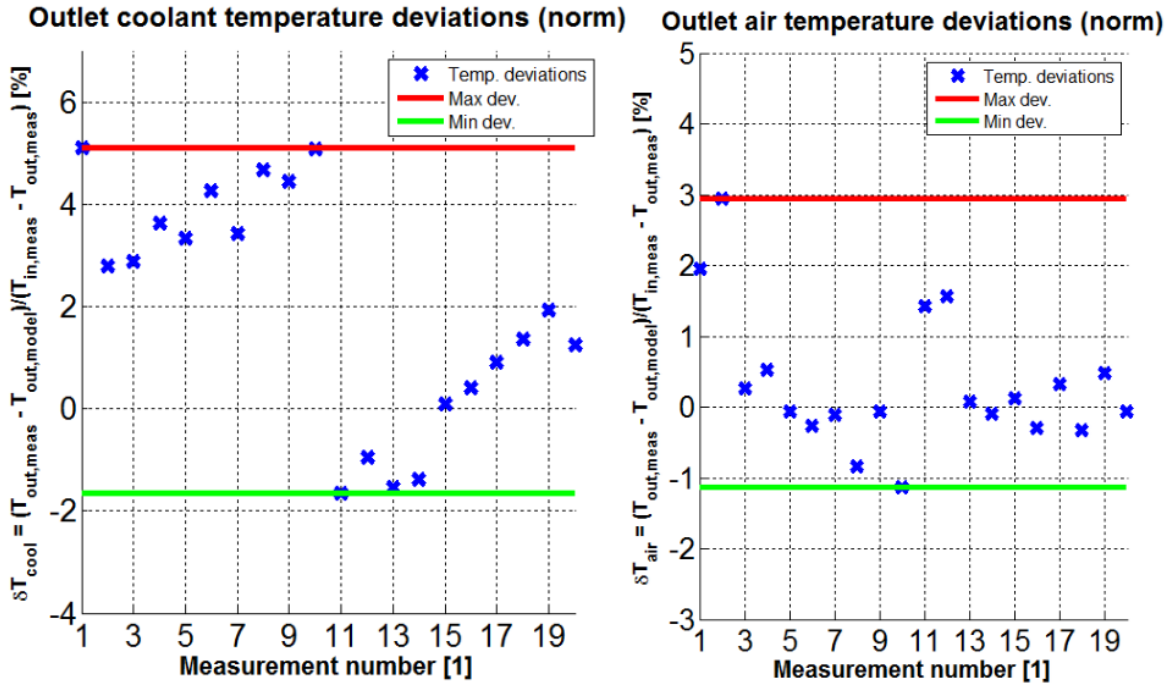


Fig. 32 Validation results for stage 1

Validation is valid in range

$$\dot{m}_g \in \langle 0.0145, 0.077 \rangle \text{ kg/s},$$

$$\dot{m}_c \in \langle 0.134, 0.201 \rangle \text{ kg/s},$$

$$T_{g,in} \in \langle 39, 73 \rangle \text{ °C},$$

$$T_{c,in} \in \langle 10, 13 \rangle \text{ °C}.$$

The average error in outlet air temperature is 2.4% and error in outlet coolant temperature is 2.6%. This is a very good agreement, but further verification is performed. Additional 5 points are measured on different heat exchanger (same type, but different dimensions). No increase in model error is observed, so proper choice of effectiveness coefficients can be concluded. No rapid increase in model error is observed on the boundary of validity range, so reasonable extension of this range is possible.

#	$T_{g,0}$ (°C)	$T_{g,out}$ (°C)	$T_{c,0}$ (°C)	$T_{c,out}$ (°C)	$\dot{m}_g$ (kg/s)	$\dot{m}_c$ (kg/s)	$w_{v,0}$ (1)	ΔH_g (W)	ΔH_c (W)	HB_{err} (%)
1	40.14	12.94	11.91	12.65	0.016	0.134	0.006	430.29	418.50	2.73
2	40.61	12.66	11.30	11.84	0.016	0.201	0.007	457.00	458.61	-0.37
3	40.36	13.58	11.77	13.19	0.030	0.134	0.006	819.08	801.49	2.15
4	40.36	13.03	11.16	12.13	0.030	0.200	0.008	838.50	815.81	2.70
5	39.99	14.90	11.84	13.85	0.046	0.135	0.006	1164.51	1132.36	2.76
6	40.26	14.18	11.18	12.57	0.045	0.201	0.007	1213.23	1165.50	3.93
7	40.38	16.16	11.82	14.43	0.061	0.134	0.006	1507.28	1464.60	2.83
8	40.44	15.33	11.25	13.02	0.061	0.201	0.007	1565.52	1490.73	4.78
9	39.80	17.21	11.89	14.90	0.076	0.134	0.006	1752.39	1688.10	3.67
10	40.15	16.44	11.36	13.46	0.077	0.201	0.007	1866.26	1766.33	5.35
11	69.27	13.42	11.82	13.33	0.014	0.135	0.008	829.85	855.41	-3.09
12	68.75	12.91	11.26	12.28	0.015	0.199	0.008	829.91	850.82	-2.53
13	71.19	14.57	12.07	14.65	0.025	0.136	0.006	1447.34	1473.14	-1.79

14	71.29	13.60	11.24	13.03	0.025	0.199	0.007	1477.52	1498.03	-1.39
15	69.04	17.13	12.07	15.88	0.040	0.135	0.006	2150.15	2156.33	-0.29
16	70.61	16.09	11.24	13.94	0.040	0.199	0.007	2261.97	2250.11	0.52
17	69.80	19.93	12.14	17.23	0.057	0.135	0.007	2885.02	2875.82	0.32
18	70.46	18.63	11.36	14.88	0.056	0.199	0.007	2973.64	2931.74	1.41
19	69.35	21.85	12.07	18.05	0.070	0.134	0.007	3404.72	3366.52	1.12
20	72.31	21.09	11.45	15.81	0.070	0.199	0.007	3671.76	3635.40	0.99

Table 3 Measurement points for 1. stage of validation

3.10.3 Stage 2 validation

The second stage of validation is performed with condensation on the air side and R134a as a refrigerant and its main target is finding coefficients S_p and S_t for theoretical mass transfer coefficients. Multiple sets of measurements (70 points) are used for this validation, but only the selection of points (44 points) is used for model calibration (heat balance error <10% required). High sensitivity on input data is observed, mainly in refrigerant mass flow. A similar approach using *fminsearch* is applied and optimal parameters are found. Target (fitness function) used for model validation

$$TF_2 = \sum_{i=1}^{n} \left[\begin{array}{c} w_g \cdot \frac{\left\| T_{g,out,meas,i} - T_{g,out,model,i} \right\|}{T_{g,out,meas,i}} + w_c \cdot \frac{\left\| T_{c,out,meas,i} - T_{c,out,model,i} \right\|}{T_{c,out,meas,i}} + \\ w_{film} \cdot \frac{\left\| \dot{m}_{film,meas,i} - \dot{m}_{film,model,i} \right\|}{\dot{m}_{film,meas,i}} + w_{qa} \cdot \frac{\left\| \Delta H_{air,meas,i} - \Delta H_{air,model,i} \right\|}{\Delta H_{air,meas,i}} \\ + w_{qc} \cdot \frac{\left\| \Delta H_{ref,meas,i} - \Delta H_{ref,model,i} \right\|}{\Delta H_{ref,meas,i}} \end{array} \right]. \quad (94)$$

This function has a similar form as TF_1. (equation (91)) Additional terms represent a relative error in condensate mass flow and total rate of heat transfer. Due to measurement errors, the total rate of heat transfer evaluated from parameters on the refrigerant side is different from the rate evaluated from parameters on the air side. Higher precision of parameters on refrigerant is expected, so weights are chosen accordingly. Using optimization procedures, minimum of function TF_2 is found and optimal corrections for theoretical mass transfer coefficients are identified as

$$S_p = 0.31, \quad (95)$$

$$S_t = 0.31. \quad (96)$$

Valid for weights $w_g = 0.5, w_c = 0.5, w_{film} = 1, w_{qa} = 0.4, w_{qc} = 1.2$. In this case, another simplification is applied by assuming

$$S_p = S_t, \quad (97)$$

This speeds up optimization process very much at a minimal cost of model precision. Results from calibration are in Fig. 33 – comparison of the normalized rate of heat transfer on air side (ratio of air enthalpy change to inlet air enthalpy) and normalized condensate mass flow (ration of condensation mass flow to inlet mass flow of air), evaluated from measurement and model results. Measurements points are also divided into 2 groups: *SAT* – inlet air state is close to saturation (RH > 94%) and *NON-SAT* – RH <= 94%.

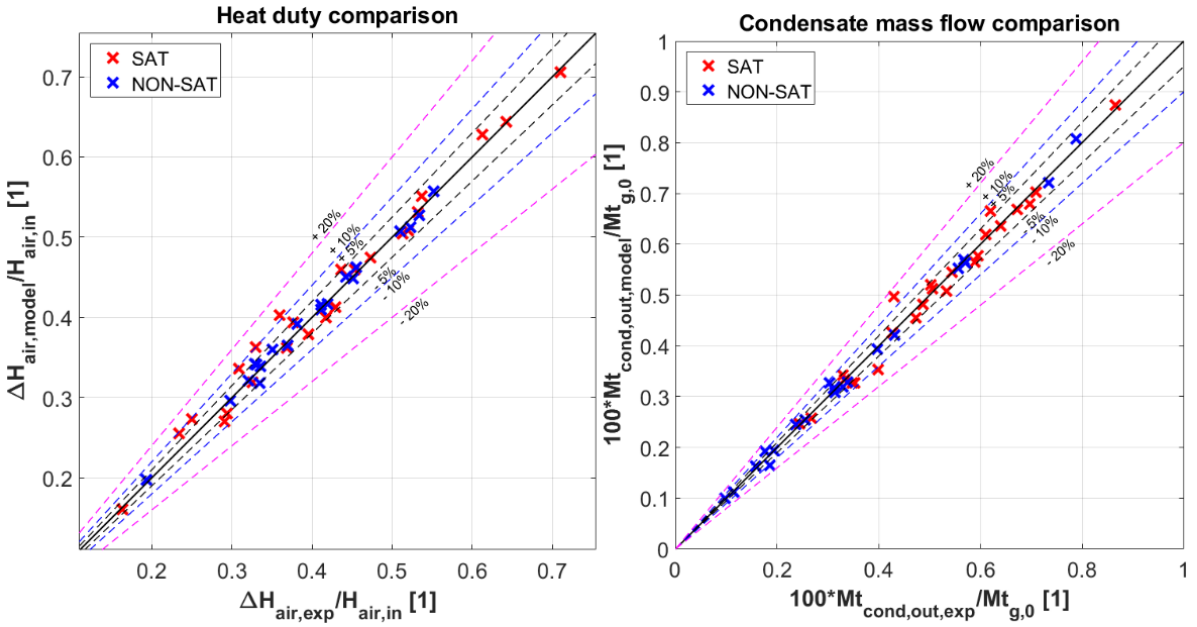


Fig. 33 Results from stage 2 model validation

Validation is valid in range

$$\dot{m}_g \in \langle 0.0156, 0.111 \rangle \text{ kg/s},$$

$$\dot{m}_{ref} \in \langle 0.004, 0.009 \rangle \text{ kg/s},$$

$$T_{g,in} \in \langle 30, 71 \rangle \text{ °C},$$

$$T_{ref,in} \in \langle 0.5, 24 \rangle \text{ °C}.$$

and additional refrigerant parameters

$$p_{evap} \in \langle 292, 629 \rangle \, kPa,$$

$$p_{cond} \in \langle 1073, 2589 \rangle \, kPa.$$

Total heat duty predicted by the model agrees with experimental results within 4.6%, and condensate mass flow within 4.2%, which is very good agreement. For the same reason, as in Stage 1, range of validity can be reasonable extended.

3.10.4 Stage 3 validation

The third stage of validation is focused on “data transfer” from the MATLAB model into commercial CFD. The main goal is to verify agreement of results from MATLAB and CFD simulation and minimize error caused by data transfer. This error is mainly caused by linear interpolation of MATLAB results on a mesh in CFD SW. In this stage, condensation on air side is present and R134a is used as a refrigerant. Total of 4 fields are transferred - $\dot{Q}_{vol,fin}, \dot{Q}_{surf,tube}, \dot{M}_{vol,fin}, \dot{M}_{surf,tube}$. Fields $\dot{Q}_{vol,fin}$ and $\dot{M}_{vol,fin}$ simulate the presence of fins and are implemented in for of “volumetric sink field”. Fields $\dot{Q}_{surf,tube}$ and $\dot{M}_{surf,tube}$. simulate the presence of tubes and are implemented in for of “heat/mass flux field”.

Transfer of volumetric heat/mass sources is validated with following example. MATLAB simulation is completed, while heat transfer coefficient on tubes is forced to 0, this represents heat exchanger composed only from fins with given temperature distribution. (high water mass

flow on coolant is used to maximize heat transfer and mass transfer coefficient is artificially increased to maximize mass transfer). Inlet conditions for testing case are $\dot{M}_g = 0.08kg/s$, $\dot{M}_c = 0.19kg/s, T_{c0} = 45℃, T_{g,0} = 67.2℃, w_{v,0} = 0.12$. After MATLAB simulation is successfully completed, data are transferred to CFD SW and implemented into source term for energy equations. Results from CFD simulation are shown in Fig. 34.

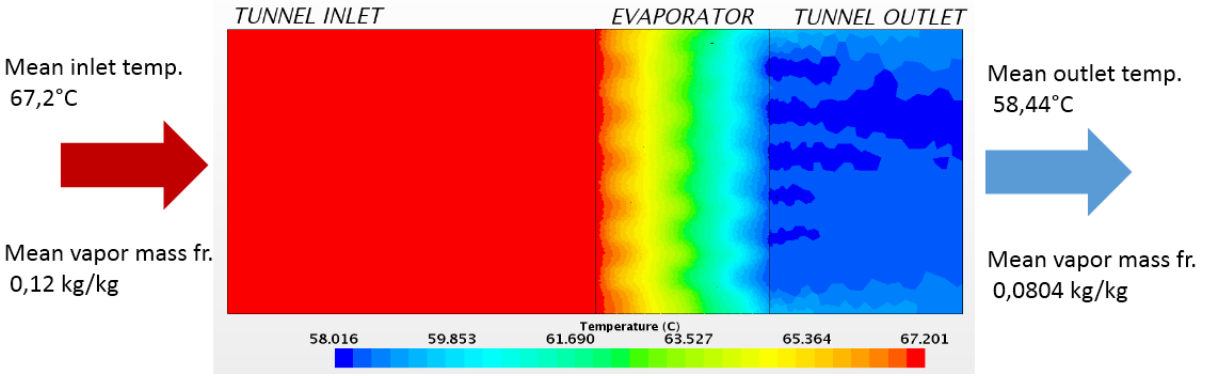


Fig. 34 Temperature field from CFD simulation with volumetric heat sinks

MATLAB and CFD results are compared in Fig. 35. Average air temperature along x direction (flow direction) in evaporator region is analyzed.

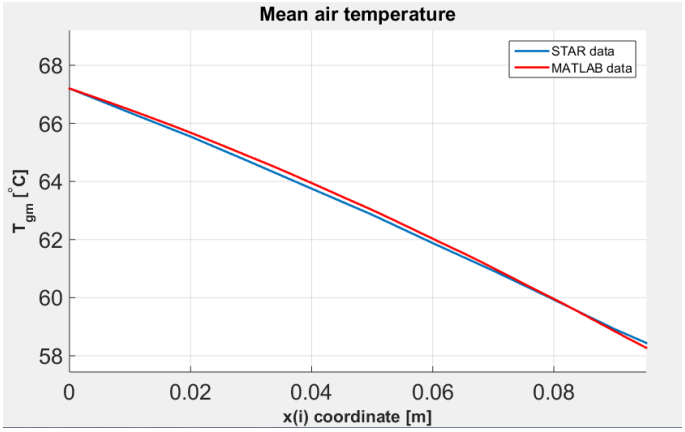


Fig. 35 Average temperature comparison

Mass fraction field from CFD simulation is shown in Fig. 36.

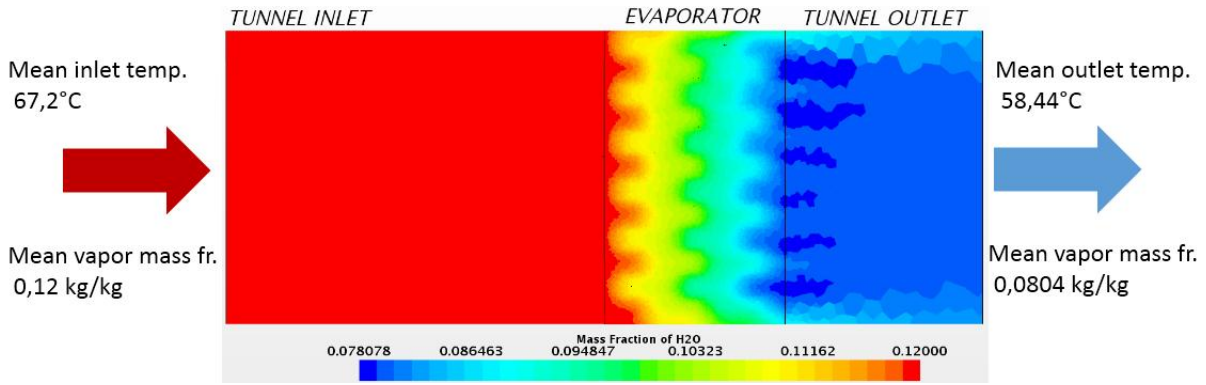


Fig. 36 Mass fraction field from CFD simulation with volumetric mass sinks

Average humidity (water vapor mass fraction) along x direction (flow direction) in evaporator region is analyzed and compared to MATLAB results in Fig. 37.

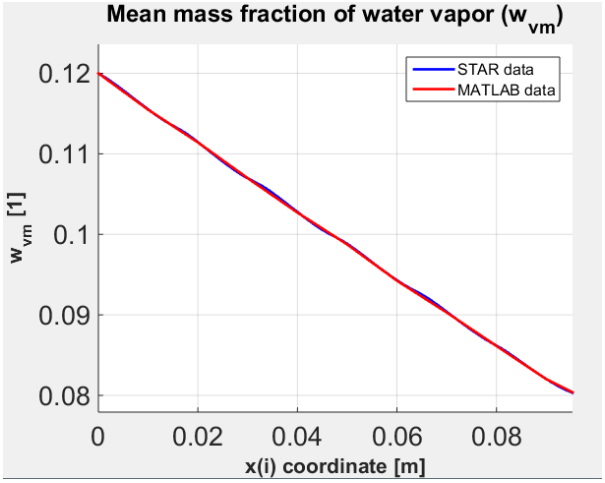


Fig. 37 Average mass fraction comparison

Comparison of average outlet parameters shows a minimal error of data transfer from MATLAB to CFD.

	STAR CCM	MATLAB	Error 1[1] (%)	Error 2[2] (%)
Temp. (°C)	58,44	58,27	0,3	1,9
Vapor mass fr.(kg/kg)	0,0804	0,0803	0,12	0,25
Transferred heat[3] (W)	756,3	760,9	-0,6	-0,6

Table 4 Comparison of outlet parameters

Sensitivity analysis of MATLAB mesh is also performed to verify data transfer, arbitrary inlet conditions are used.

MATLAB #elements		MATLAB results			CFD results			Error		
NxM	Elems	$T_{g,out}$ (°C)	$w_{v,out}$ (1)	Q_{air} (W)	$T_{g,out}$ (°C)	$w_{v,out}$ (1)	Q_{air} (W)	Temp (%)	Humid (%)	Q_{air} (%)
179 x 291	52089	58.25	0.08020	762.0	58.44	0.0803	756.0	-0.33	-0.12	0.79
154 x 251	38654	58.27	0.08030	761.0	58.47	0.0805	754.1	-0.34	-0.25	0.91
129 x 210	27090	58.29	0.08050	760.1	58.49	0.0807	751.9	-0.34	-0.25	1.08
112 x 182	20384	58.33	0.08070	758.3	58.53	0.0810	748.6	-0.34	-0.37	1.28
73 x 119	8687	58.40	0.08175	756.2	58.62	0.0821	741.7	-0.38	-0.43	1.92

Table 5 Data points for sensibility analysis of MATLAB mesh on data transfer

A sufficient number of elements in MATLAB model is ca.30 000. (for given geometry) Further mesh improvement doesn't provide much higher precision (a difference between MATLAB and CFD), but significantly increases computation time (an increase from 4 min to 10 min) in MATLAB. A similar analysis is done with 3 other inlet conditions, but alike results are obtained. Biggest deviation is in total transferred heat (integral volumetric heat sinks over evaporator domain) and is not exceeding 1.9% (Fig. 38).

[1] Error 1 – difference at outlet/MATLAB outlet
[2] Error 2 – difference at outlet/abs(MATLAB inlet – MATLAB outlet)
[3] Without latent heat of condensation

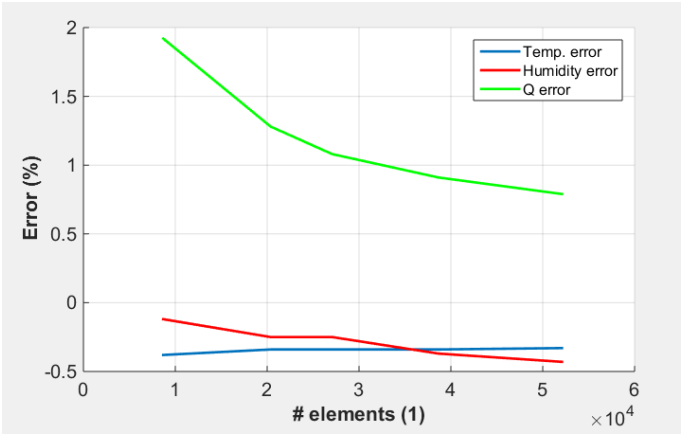


Fig. 38 Sensibility analysis of MATLAB mesh on data transfer

Only volumetric heat/mass sources are applied in the previous section. In order to verify transfer of heat/mass fluxes, tube surfaces (and tube bundle itself) need to be defined in evaporator regions. Java scripts (described in more detail in chapter 4.2) are used to easily generate these tube surfaces (identified as boundaries) and assign appropriate values of heat and mass fluxes. A precision of data transfer of these fluxes (volumetric heat sources are excluded) is performed for multiple inlet conditions and geometrical parameters and error of outlet parameters (air temperature, humidity and total rate of heat transfer) does not exceed 0.2%.

In the final step, simultaneous data transfer of both types of fields is verified. Developed procedure (chapter 3.12 and 4) based on Java scripts and macros is used for this. A total of 20 different inlet conditions and geometrical parameters is applied and error (a difference between MATLAB and CFD result) of outlet parameters is evaluated. A maximal error does not exceed 1.9%, which is same as for volumetric heat sinks only. From this fact, it can be concluded that majority of overall error is caused by imprecise interpolation of volumetric sinks. An example of temperature field from CFD simulation is in Fig. 39. (inlet conditions: $T_{g0} = 45℃$, $T_{c0} = 22$, $\dot{M}_{g0} = 0.1\ kg/s$, $\dot{M}_{c0} = 0.1\ kg/s$, $w_{v0} = 0.00078\ kg/kg$)

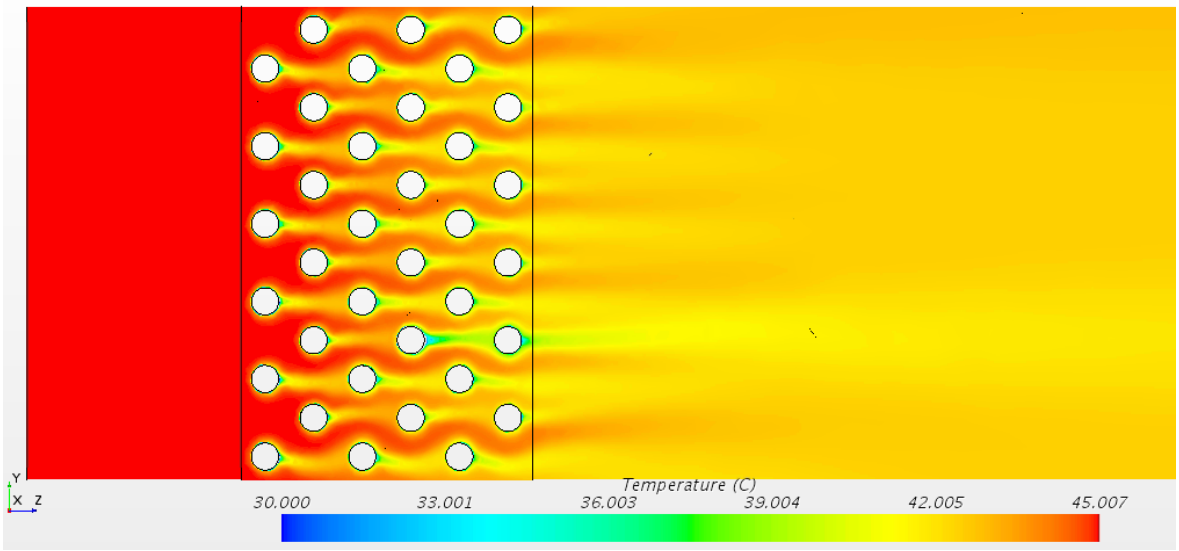


Fig. 39 Temperature field from CFD simulation

One problem with data interpolation is discovered after tubes are added into evaporator domain– heat/mass sinks fields interpolated in CFD are distorted close to tube surface.

Additional interpolation at tube cross-section in MATLAB model is added to improve precision of data transfer. Comparison of previous and improved interpolation method for arbitrary inlet conditions is shown in Fig. 40.

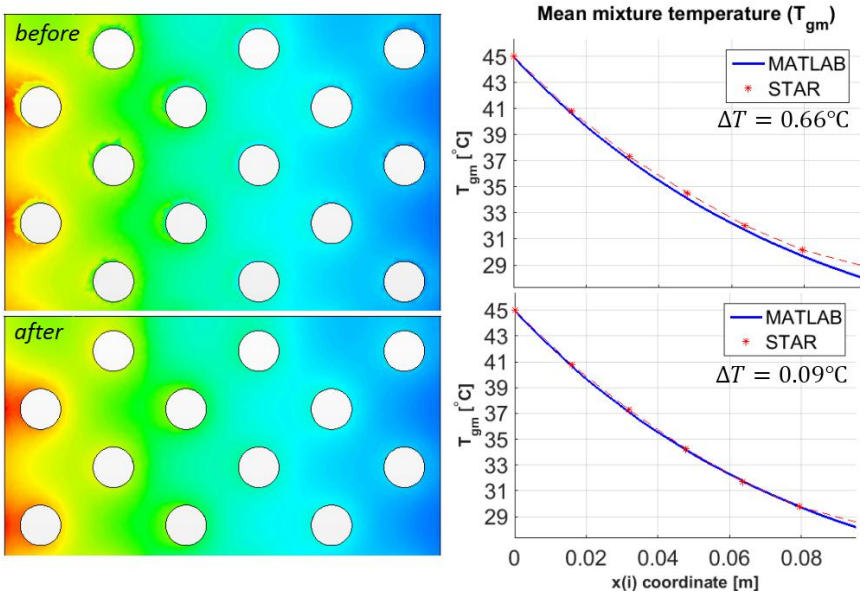


Fig. 40 Sink field repair

The overall precision of data transfer is verified and corrected in this chapter. Very good agreement of all important parameters between MATLAB and CFD simulation is achieved with an error not exceeding 1.9%. The main cause of this error is a linear interpolation of MATLAB results (3D fields) on CFD mesh, which is done with built-in functions, that cannot be easily modified.

3.10.5 Stage 4 validation

Final validation stage is focused on verification of MATLAB model with non-homogenous inlet conditions. To have well defined inlet conditions, simple partial blockage of frontal area applied in **horizontal** and **vertical** direction is used. Comparison of simulation results from MATLAB and CFD model to experimental data is also done.

Vertical blockage of 20, 40 and 60% frontal is in CFD modeled as cut-out of evaporator and outlet region, because fins don't allow air flow in z direction (perpendicular to fins). MATLAB model is modified by decreasing number of fins and total width of heat exchanger.

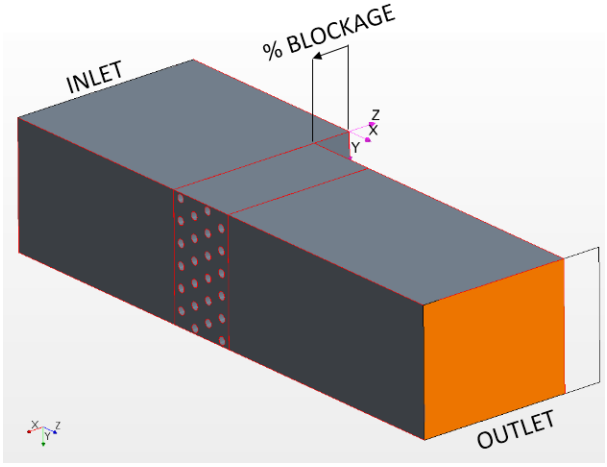


Fig. 41 Vertical blockage modeled in CFD

Total of 9 points are measured (3 different inlet conditions for every blockage), analysis of 3 selected points is shown in Table 6.

	Experiment			MATLAB			CFD		
Inlet air temp. (°C)	39,7	39,8	39,9	39,7	39,8	39,9	39,7	39,8	39,9
Air mass flow (g/s)	52,4	52,1	51,9	52,4	52,1	51,9	52,4	52,1	51,9
Inlet RH (%)	96	96	96	96	96	96	96	96	96
Outlet air temp. (°C)	36,30	36,52	36,71	36,32	36,49	36,57	36,27	36,57	36,42
Outlet mass fr. (1)	0,0395	0,0400	0,0410	0,0395	0,0397	0,0400	0,0394	0,0397	0,0399
Condensate flow (g/s)	0,387	0,382	0,383	0,371	0,368	0,367	0,366	0,369	0,374
Heat duty (W)	1095	1086	1078	1099	1087	1085	1089	1078	1070
Latent heat (W)	933	920	923	894	888	885	903	889	869
Sensible heat (W)	162	166	155	205	199	200	186	189	201
HB error (%)	1,5	0,7	1,2	0,4	0,2	0,9	→ 0	→ 0	→ 0
Blockage (%)	20	40	60	20	40	60	20	40	60

Table 6 Selected results of vertical blockage simulation and comparison to exp. data

The very small difference between numerical simulations and experimental results is observed, not exceeding 3,2% for condensate mass flow. This error is insignificant, considering an error of data transfer and measurement precision (condensate mass flow is measured as a time integral of collected condensate in a graduated cylinder during 15 min interval). Velocity field for 60% of vertical blockage is shown in Fig. 42 as an example.

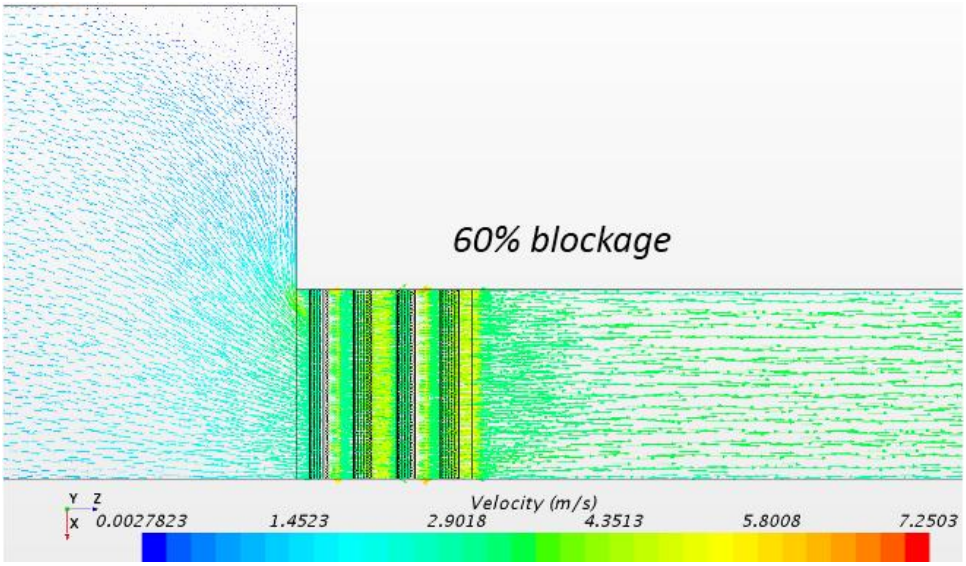


Fig. 42 Velocity field from CFD simulation for 60% of blockage

An interesting conclusion can be made from temperature field comparison. It is obvious, that even with 40% of heat transfer are blocked, the refrigerant temperature reaches inlet air temperature in the first column of tubes, so evaporator is for selected inlet conditions very oversized. For 60% of blockage, the second column of tubes gets slightly involved in heat transfer.

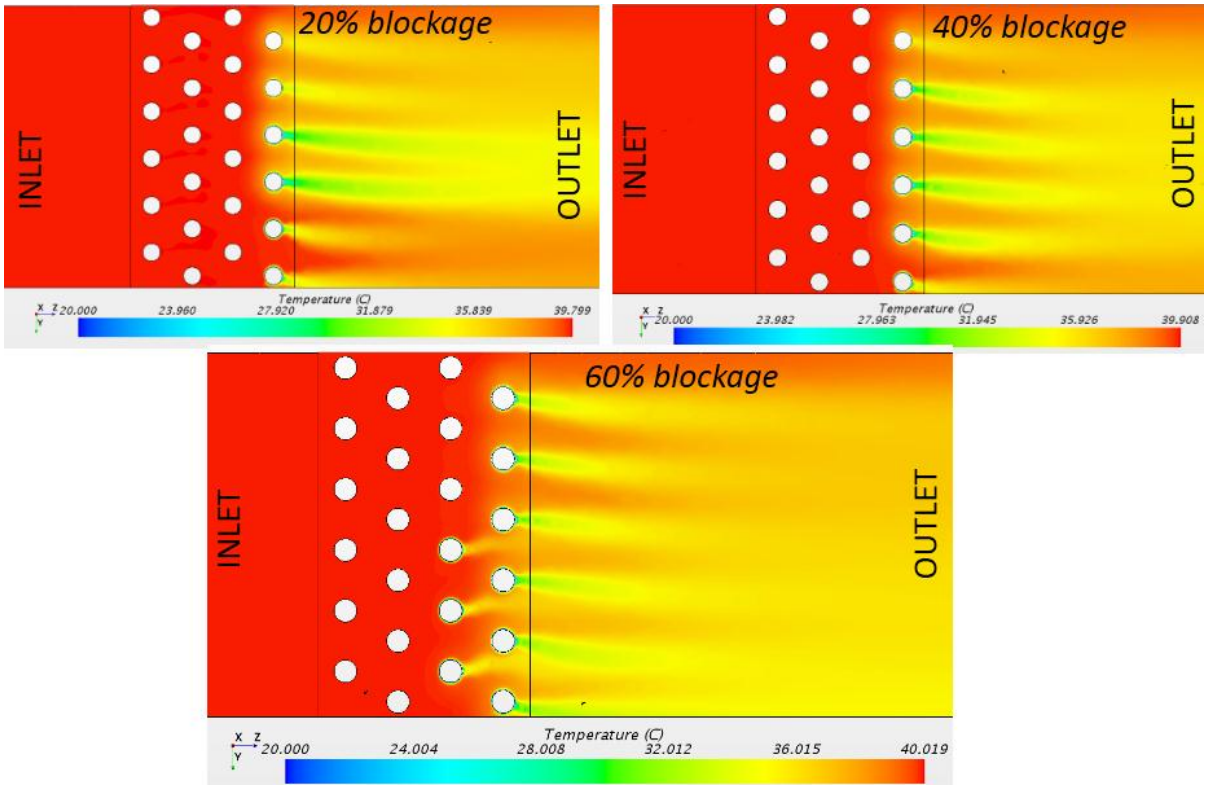


Fig. 43 Temperature fields from CFD simulation for 20, 40, 60% of vertical blockage

Simplified **horizontal blockage** is modeled as a baffle in front of evaporator domain, covering 20, 40 and 60% of the frontal area. A thickness of baffle is 2 mm and no modifications are made to MATLAB model (all dimensions are the same).

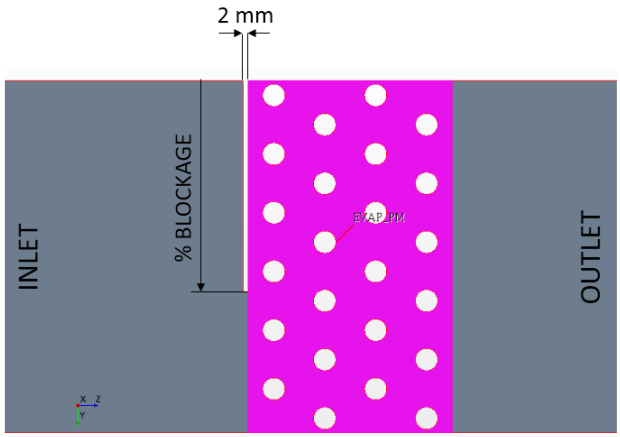


Fig. 44 Horizontal blockage modeled in CFD

Another 9 points with similar inlet conditions are measured on the test rig and results are analyzed. Comparison of 3 selected points is done in Table 7.

	Experiment			MATLAB			CFD		
Inlet air temp. (°C)	39,7	39,8	39,9	39,7	39,8	39,9	39,7	39,8	39,9
Air mass flow (g/s)	52,4	52,1	51,9	52,4	52,1	51,9	52,4	52,1	51,9
Inlet RH (%)	96	96	96	96	96	96	96	96	96
Outlet air temperature (°C)	36,30	36,52	36,71	36,35	36,44	36,56	36,42	36,53	36,73
Outlet vapor mass fraction (1)	0,0395	0,0400	0,0410	0,0396	0,0398	0,0401	0,0392	0,0396	0,040
Cond. flow (g/s)	0,387	0,382	0,383	0,370	0,368	0,363	0,368	0,365	0,363
Heat duty (W)	1095	1086	1078	1094	1088	1076	1084	1074	1075
Latent heat (W)	933	920	923	891	886	876	898	890	894
Sensible heat (W)	162	166	155	203	202	200	186	184	181
HB error (%)	1,5	0,7	1,2	0,1%	0,3%	0,1%	→ 0	→ 0	→ 0
Blockage (%)	20	40	60	20	40	60	20	40	60

Table 7 Selected results of horizontal blockage simulation and comparison to exp. data

Almost no difference in temperature fields from CFD simulations is observed, which is expected, because same heat/mass sink fields from MATLAB simulation are used. From air temperature distributions is obvious that evaporator is for selected inlet conditions oversized, because all refrigerant is vaporized within the first column of tubes, so no heat exchange occurs in a large portion of the evaporator, which agrees with results from **vertical blockage** measurement. A similar maximal error of 3,1% in condensate mass flow is observed. An example of temperature fields for 20, 40 and 60% of blockage is in Fig.45.

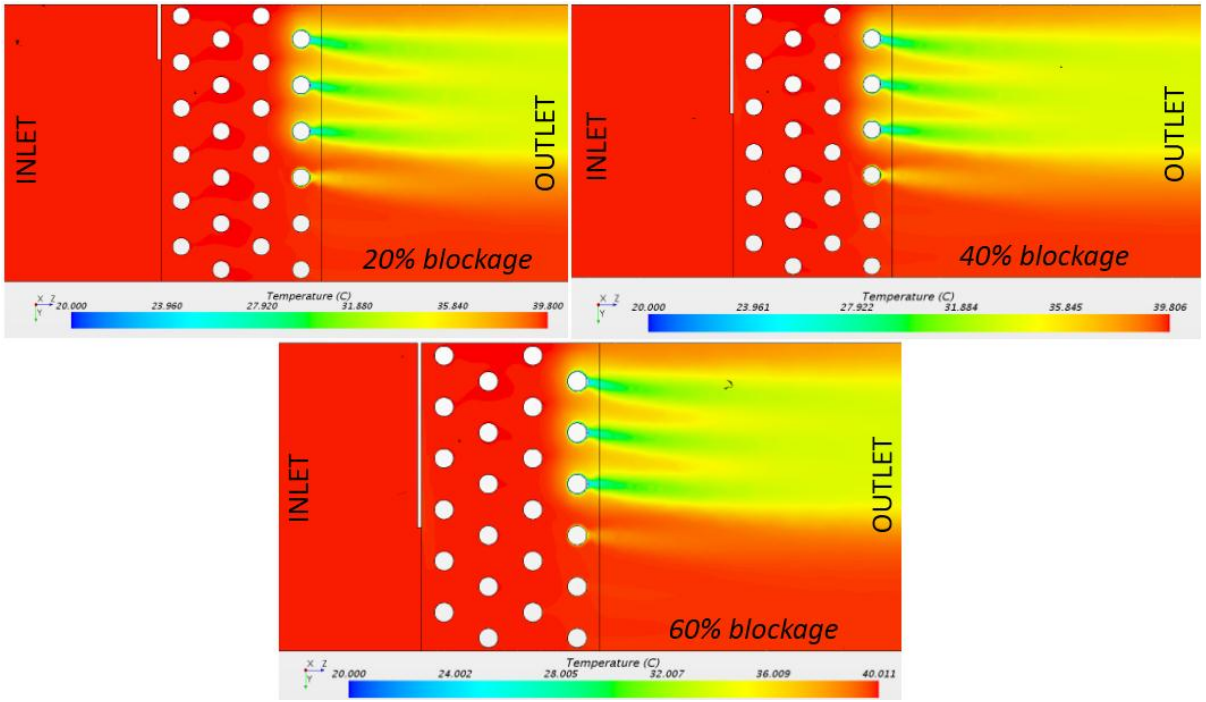


Fig. 45 Temperature fields from CFD simulation for 20, 40, 60% of horizontal blockage

3.11 Model usage

Following chapters should be used as a short manual that helps setting up the model, running simulation and processing results. Model is used as a standard MATLAB application

(.m file* or MATLAB *gui file*), so MATLAB SW is required. (version R2014b or later) No previous knowledge of MATLAB or other programming languages is required prior using the model.

3.11.1 Working folder description

All files and folders located in model root directory are described in *Appendix 1.*

3.11.2 Text/*.m file interface

The first developed and more basic version of MM is "text file" version – all solver parameters, initial conditions and geometrical parameters are controlled and modified through "text files" (*.m files – MATLAB native files) located in "*Params*" folder. This is standard approach to control variables and parameters of MATLAB applications, that is straightforward for more experienced users and provides direct control for more advanced parameters of the model.

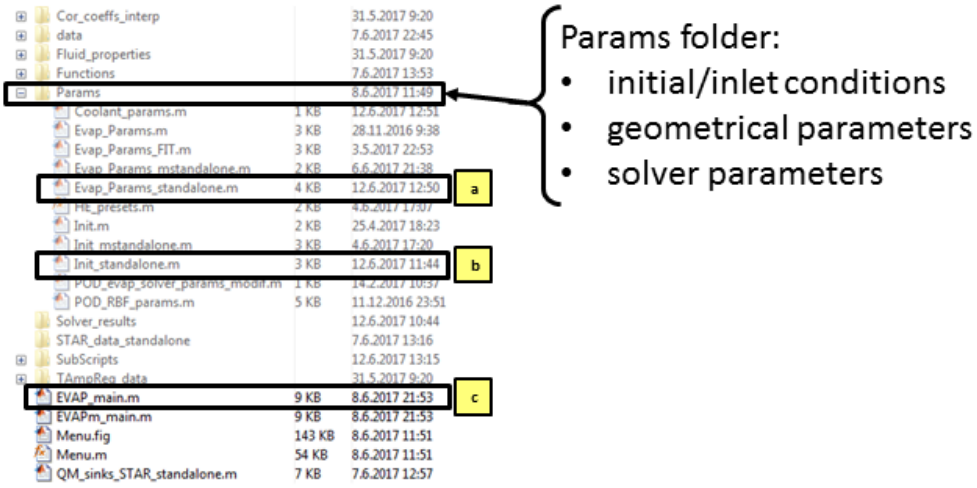


Fig. 46 Params folder content

Description of files "*Evap_Params_standalone.m*" and "*Init_standalone.m*" is in *Appendix 2*.

3.11.3 Menu interface

Model is also equipped with user menu (developed in MATLAB GUI tool) for a more user-friendly environment, that should be used by less experienced MATLAB users, to avoid possible errors and mistakes. The menu is started by running "*Menu.m*" in working folder.

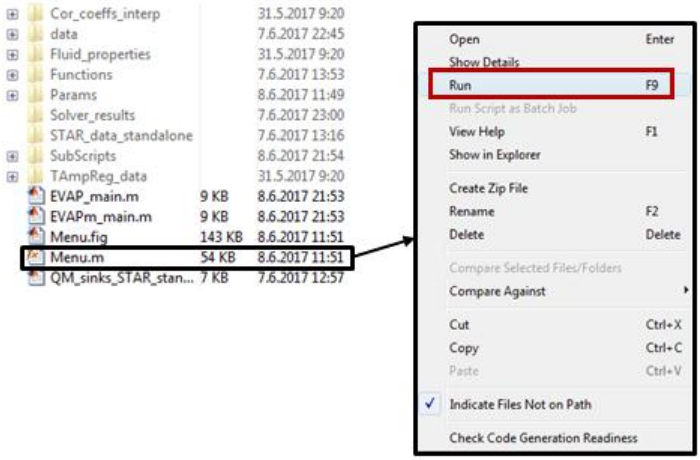


Fig. 47 Running model with menu interface

Description of menu interface is in *Appendix 3*.

3.11.4 Simulation procedure

Simulation procedure using MATLAB model is similar to other numerical simulations – initial run with a coarse mesh, finding optimal initial parameters and choosing stable solver, final run on a fine mesh and data post-processing. The code is optimized for a 1 core application (except built-in MATLAB functions that can handle multiple cores), so using it on laptop or PC is recommended. A strategy of gradually increasing mesh resolution is recommended, because the combination of a high number of iterations ($KMit$) and very fine mesh can lead to very high overall computation time without guaranteed convergence. (See Fig. 48)

Average Iteration time is analyzed using 4-core CPU, 3,3 GHz, 64bit, 8GB DDR2 RAM machine.

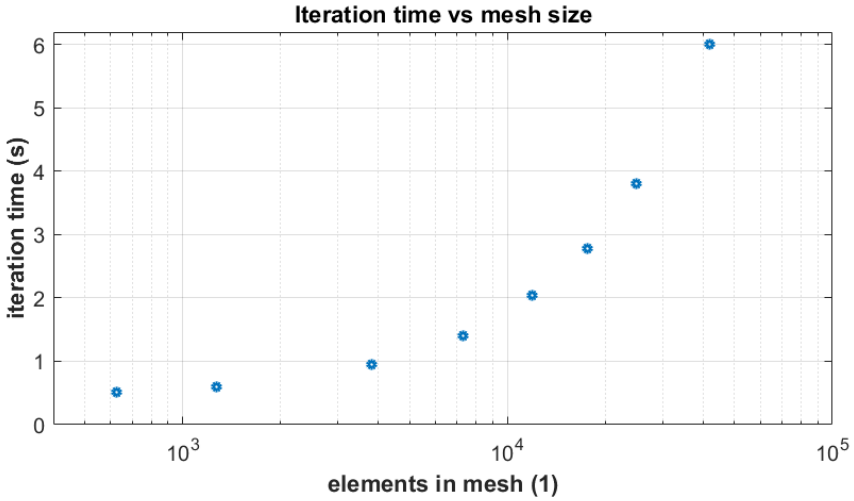


Fig. 48 Iteration time vs mesh size

Recommended steps of simulation in MATLAB model with "menu" interface:

1. Start user menu ("*Menu.m*") and specify inlet/geometry parameters.
2. Solver selection and simulation testing run:
    a. for the first run using **FORCED** solver is recommended (not required) to verify that $Q_{fr} > 1 \rightarrow$ all refrigerant is evaporated. Do not save results (43) nor create sinks for STAR-CCM+(44). (recommended number of iterations *KMit(12) ~ 30*)
    b. For the second run use **STANDART** solver without REDISTRIBUTION CORRECTION to check if solution is stable. (see note 9) Do not save results (43) nor create sinks for STAR-CCM+(44). (recommended number of iterations *KMit(12) ~ 200*)
    c. If solution from previous steps is stable, use REDISTRIBUTION CORRECTION to minimize HB error. (recommended number of iterations *KMit(12) ~ 200*). If solution from previous step is not stable, use **CAREFUL** solver. (recommended number of iterations *KMit(12) ~ 300*)
3. If HB error from point c is not sufficient, use final value of *URFqMod(54)* as an initial value for next run. (recommended number of iterations *KMit(12) ~ 100-300*)

4. If HB error from previous point is sufficient, use final value of *URFqMod(54)* as an initial value and disable RC modification by setting *dURFq(54) = 0*, *HBLim(57) = 0*, *dURFqLim(58) = 0*. At this point you should create sources for STAR-CCM+ (44) and also save results (43). (recommended number of iterations *KMit(12) ~ 100*)

3.11.5 Solver choice

MATLAB model is equipped with 3 main solvers – *STANDART*, *CAREFUL*, *FORCED*. STANDART solver follows diagram if Fig. 18 and provides the fastest simulation with high precision, but in some cases, mainly if refrigerant gets superheated very quickly (for example the refrigerant temperature in 3rd tube section is equal to inlet air temperature), *STANDART* solver will be unstable (of keeps oscillating between 2 modes) and will not give reliable results. The error message may also appear in MATLAB Command window after couple iterations:

```
-----------------------------------------------------------------
Bad number of roots! nR = 0 mtf_ji = 1.3549e-11+4.8249e-19i dmtf_ji
y = 0.07377307 wv = 0.04671000 wvw0 = 0.04591975-0.00000000i yvpsat
-----------------------------------------------------------------
```

Fig. 49 Error message triggered by simulation instability

In this case, one option is to decrease under relaxations factors (*URF* for refrigerant temperature and surface temperature distribution), but using *CAREFUL* solver is recommended. Usually, this solver is slower, because the algorithm is based on optimization of 2 parameters - Q_{fr} (desired value $Q_{fr} = 1$) and HB_{err} (desired value $HB_{err} = 0$).

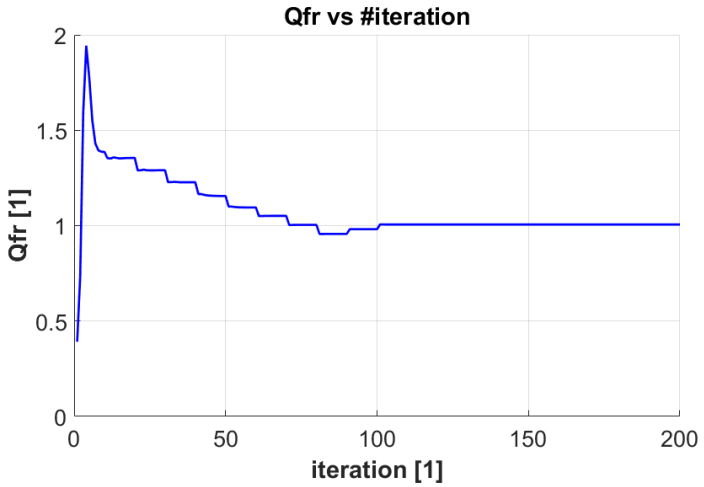


Fig. 50 Q_{fr} vs iteration number

FORCED solver can be used for quick verification that all refrigerant gets evaporated for given air inlet conditions. (if parameter $Q_{fr} \geq 1$ all refrigerant is evaporated, if $Q_{fr} \leq 1$, only $100 \cdot Q_{fr}\%$ of refrigerant is fully evaporated – for example: $Q_{fr} = 0.65 \rightarrow 65\%$ of refrigerant is vaporized in evaporator outlet)

3.11.6 Solver initiation process

After solver is initiated ("Solve HE!" button in user menus pressed or "EVAP_main.m" is executed), evaporator parameters and initial conditions are loaded, meshing is performed (sub-optimization based on minimization of tube cross section error), after that, global stiffness matrix for FEM method is assembled and LU factorization is performed. Multiple messages appear in *Command Windows* in MATLAB.

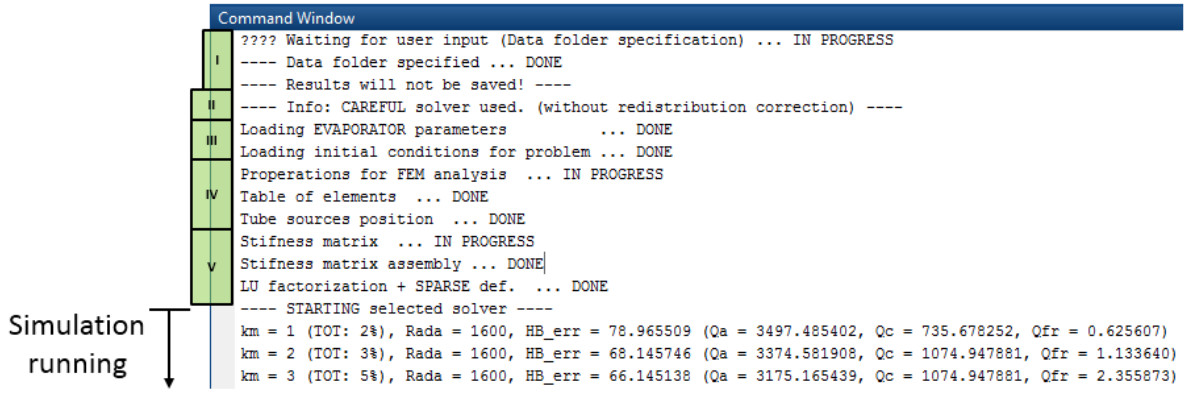


Fig. 51 Solver initiation – messages in Command Window

Description of messages:

I. User specifies folder where to save results (or not save results at all).
II. Solver used for simulation (specified in menu or in *"Evap_Params_standalone.m"*).
III. Loading inlet/initial conditions and *HE* dimensions.
IV. Meshing, creating table of tube positions, generating tube bundle and section shape.
V. Assembling stiffness matrix for *FEM* and LU factorizing.

If no errors are detected (incorrect input of solver choice), solver starts iteration process and basic monitors are displayed.

3.11.7 Solution stability

A required property of solution (of numerical methods) to achieve convergence is its stability, so it is important to monitor this property. Stability can be checked online during simulation – with iterations residuals. In this model, temperature residuals are defined as

$$\delta T_j^{(i)} = \left| \frac{max\left(\boldsymbol{T}_j^{(i+1)} - \boldsymbol{T}_j^{(i)}\right)}{\overline{\boldsymbol{T}^{(i)}}_j} \right|, \tag{98}$$

where *i* – iteration number, *j* – temperature field (air, fin, refrigerant) and humidity residuals as

$$\delta w_v^{(i)} = \left| \frac{max\left(\boldsymbol{w}_v^{(i+1)} - \boldsymbol{w}_v^{(i)}\right)}{\overline{\boldsymbol{w}^{(i)}}_v} \right|. \tag{99}$$

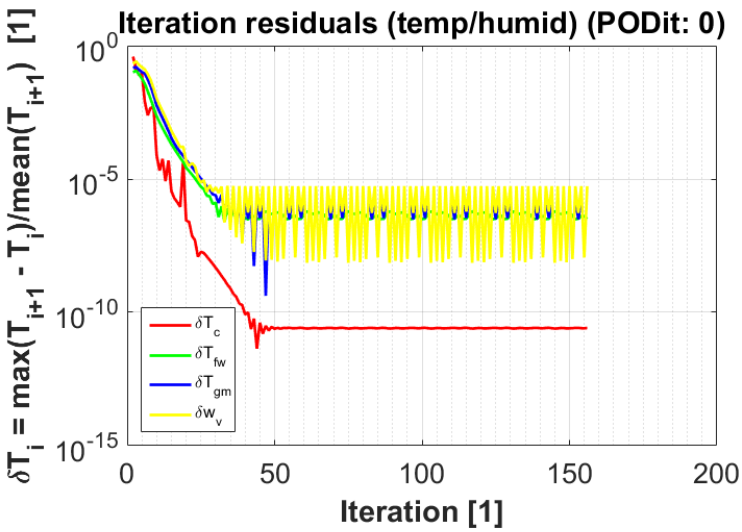


Fig. 52 Iteration residuals

The second option to check stability is to monitor heat balance error and refrigerant temperature distribution. Heat balance error is defined as

$$HB_{err} = 100 \cdot \frac{\Delta\dot{H}_g - \Delta\dot{H}_{ref} - \dot{H}_{film}}{\Delta\dot{H}_g}, \tag{100}$$

where enthalpy change on air side is equal to

$$\Delta\dot{H}_g = \dot{H}_{g,in} - \dot{H}_{g,out} = \tag{101}$$

$$= \dot{m}_{g,0}\left(w_{a,0}c_{p,a}T_{g,0} + w_{v,0}\left(c_{p,v}T_{g,0} + l_{23,0}\right)\right) - \dot{m}_{g,out}\left(w_{a,out}c_{p,a}T_{g,out} + w_{v,out}\left(c_{p,v}T_{g,out} + l_{23,0}\right)\right).$$

and enthalpy change on refrigerant side

$$\Delta\dot{H}_{ref} = \Delta\dot{H}_{ref,out} - \Delta\dot{H}_{ref,in} \tag{102}$$

$$= \dot{m}_{r,0}\left(c_{p,r}\big|_{T_{r,out}}\, T_{r,out} + \left(l_{23,r}\big|_{T_{r,0}} - \Delta l_{23,cond}\big|_{T_{cond,out},p_{cond}}\right) - c_{p,r}\big|_{T_{r,0}}\, T_{r,0}\right).$$

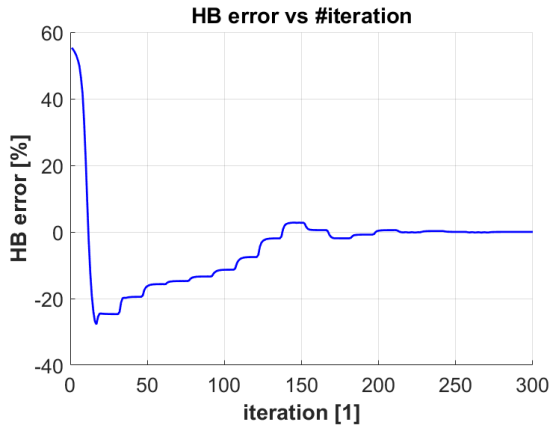


Fig. 53 Heat balance error

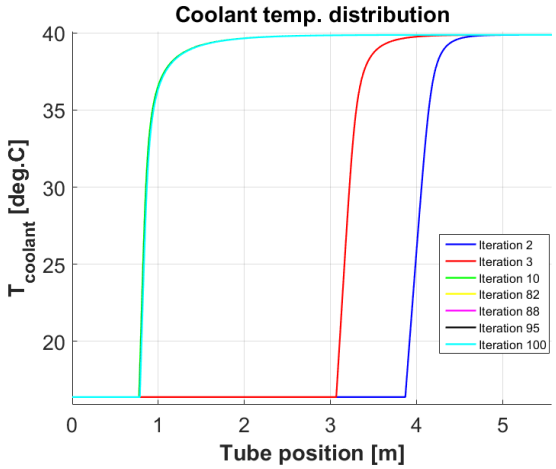


Fig. 54 Refrigerant temperature distribution

Temperature distribution should stabilize after couple iterations regardless of using *STANDART* solver settings, otherwise, *CAREFUL* solver should be used.

3.11.8 Solver completion

If there is no problem during simulation, the solver will achieve a required number of iterations ($KMit$), complete all sub-procedures (to achieve 100% completion) and will end with the following message.

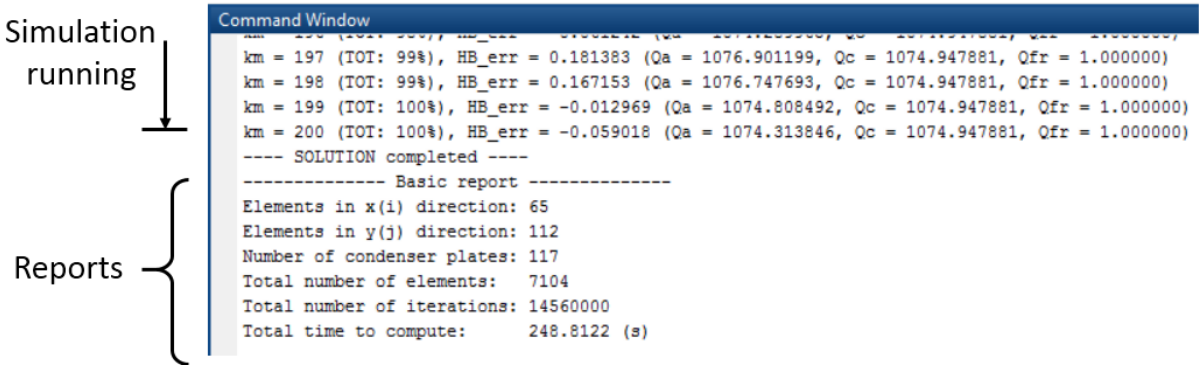


Fig. 55 Proper simulation completion

A total number of iterations may be higher than specified value if *IIAllow* parameter is set to *true* (*Iteration Increase Allowed*) – solver keeps increasing number of iterations until sufficient HB error is achieved.

After the simulation is completed, 4 main reports show in *Command Window* to provide basic information about outlet parameters, heat duty heat transfer coefficients and heat/mass sinks for CFD. List of main reports:

- Basic report (see *Appendix 4*)
- Enthalpy report (see *Appendix 4*)
- HT report (see *Appendix 5*)
- Heat and mass sinks creation (see *Appendix 5*)

Also 6 plots are displayed after successful simulation:

- Mixture (air) temperature over plate - contour plot (see *Appendix 6*)
- Mean air/coolant temperature along x direction - line plot (see *Appendix 7*)
- Mean humidity along x direction (mass fraction/RH/specific humidity) - line plot (see *Appendix 8*)
- Volume condensation rate - contour plot (see *Appendix 9*)
- Air saturation - contour plot (see *Appendix 10*)
- Iteration deviations - line plot (see *Appendix 11*)

3.12 Data transfer and POD method

3D fields generated by MATLAB are not directly compatible with CFD, so some procedure that will transform data from MATLAB to CFD code is required. A suitable option is to use *csv files, where data are stored in the format $[x, y, z, quantity]$, import this file into CFD and then use in-built interpolation functions to reconstruct original MATLAB field. Identical coordinate systems for MATLAB and CFD model is required. Minor data loss is expected due to interpolation in locations with large gradients (an area close to tube surface), but this can be improved by using mesh with higher resolution in both MATLAB and CFD, validation of precision of data transfer is performed in chapter 3.10.4. To speed up the process of data transfer, automatic MATLAB procedure is developed. After the final simulation is complete

(final run with reasonable HB error and appropriate mesh), heat and mass sinks should be created. There are 2 options for that:

1. Check “Create sources for STAR-CCM+” (item 44) in user menu
2. Create sources separately by running “*QM_sinks_STAR_standalone.m*”

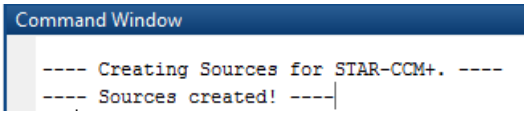


Fig. 56 Heat sources creation messages

If the first option is used, all files (see *Appendix 12*) for data transfer can be found in subfolder “STAR_data_standalone” of the folder you specified during solver initiation.

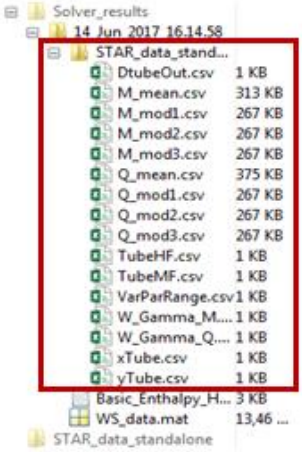


Fig. 57 Heat and mass sinks in specified folder

If the second option is used, all files can be found in “STAR_data_standalone” folder directly in working folder.

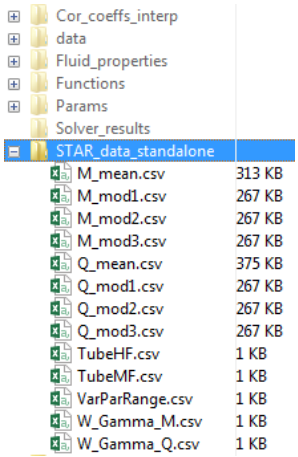


Fig. 58 Heat and mass sinks in default folder

Description of files for data transfer can be found in *Appendix 12.*

Another improvement of data transfer process is done by incorporating the discrete version of POD (*Proper Orthogonal Decomposition*) method - *Method of Snapshots*. This method allows to simplify a large amount of data into a smaller package and also to reconstruct solution for given (directly unsolved) inlet parameters very quickly. Input for POD method is set of direct MATLAB solutions for different parameters – inlet conditions.

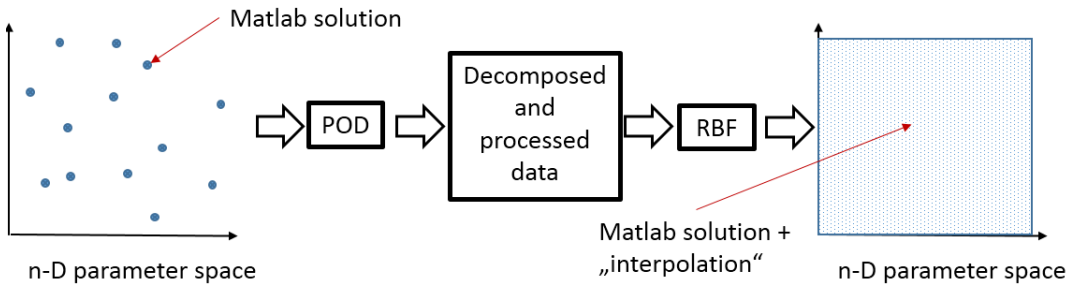


Fig. 59 POD and RBF method schema

All fields from solution are then decomposed into average values and fluctuation, which is then decomposed into modes and amplitudes (decomposition method is covered in [21]). Reconstructed solution for given inlet conditions can be written symbolically as

$$\boldsymbol{OUTPUT}_{\boldsymbol{param}_k} = \boldsymbol{OUTPUT}_{mean} + \sum_{\boldsymbol{j=1}}^{\boldsymbol{n}} \boldsymbol{MOD}_j \cdot \boldsymbol{AMPLITUDE}_{\boldsymbol{param}_k,j}, \tag{103}$$

where *n* is number of modes. Fields $\boldsymbol{OUTPUT}_{\boldsymbol{mean}}$ and $\boldsymbol{MOD}$ are constant, but $\boldsymbol{AMPLITUDE}$ is a function of inlet conditions that needs to be constructed based on direct solutions with *Radial Basis Functions* (RBF) in Gaussian form. Overall approximation function can be interpreted as RBF network.

$$h(x) = b + \sum_{k=1}^{N} h_k(x) = b + \sum_{k=1}^{N} w_k e^{-\gamma \|x - x_k\|^2}. \tag{104}$$

Where x is arbitrary points in n-D parameter space and x_k is a point for which direct simulation is performed. Coefficients b and w_k are trained with data from direct solutions by MATLAB optimization method *fminsearch*. Target (fitness) function is defined as

$$\Delta_{tot} = \sum_{k=1}^{N} \Delta(x_k), \tag{105}$$

where sample error function is evaluated as

$$\Delta(x_k) = (h(x_k) - y_k)^2. \tag{106}$$

Total approximation error function can be then written as

$$\Delta_{tot} = \sum_{k=1}^{N} \Delta(x_k) = \sum_{k=1}^{N} \left(b + \sum_{l=1}^{N} w_l e^{-\gamma \|x_k - x_l\|^2} - y_k \right)^2. \tag{107}$$

Necessary number of modes required to “precise” approximation is determined from quantity called level of correlation and it is defined as

$$L_n = \frac{\sum_{j=1}^{n} \lambda_j}{\sum_{j=1}^{N} \lambda_j};\ n = 1,2,\dots,N \tag{108}$$

Where *N* is a number of direct simulations and *n* is a number of modes, choose so $L_n > 0.99$. Implementation of POD and RBF method into the mathematical model is covered more deeply in my previous work [21].

4 CFD implementation

MATLAB model provides results quickly and easily, but in some situations, a further refined solution with more complex aerodynamical effect from complex geometry is required. For example, simulation of whole heat pump cycle with an evaporator, condenser, inlet and outlet duct. All this relates to complex geometry and non-uniformities in inlet and outlet parameters, so simplified MATLAB model cannot provide desired results. In case like this, CFD simulation is suitable, because it can handle complex geometry and flow distribution easily, but modeling heat transfer and condensation on fins and tube bundle would be very complicated, so results from simplified MATLAB model (heat and mass sinks) are used. To simulate heat transfer and condensation (mass transfer) inside evaporator within CFD SW, heat and mass sink fields need to be defined through whole evaporator region. There are 2 types of heat/mass sinks – volumetric sinks simulate heat/mass transfer on fins and surface fluxes simulate transfer of tubes. These fields are obtained with MATLAB model and are imported through *.csv files. An approach based on a porous medium is used, which allows easy implementation of sinks fields and also to define specific pressure drop (porous and viscous resistance) in all 3 directions, to model presence of fins and tubes. Coefficients for porous and viscous resistances are derived from measurements of pressure drop.

Coefficients for *Porous Inertial Resistance*. (with respect to local coordinate system)

$$PIR_{xx} = 60.6\ kg/m^4 \tag{109}$$

$$PIR_{yy} = 38.7\ kg/m^4 \tag{110}$$

$$PIR_{zz} = 0\ kg/m^4 \tag{111}$$

Coefficients for *Porous Viscous Resistance*. (with respect to local coordinate system)

$$PVR_{xx} = 37.6\ kg \cdot m^{-3} \cdot s^{-1} \tag{112}$$

$$PVR_{yy} = 37.2\ kg \cdot m^{-3} \cdot s^{-1} \tag{113}$$

$$PVR_{zz} = 10\,000\ kg \cdot m^{-3} \cdot s^{-1} \tag{114}$$

Measurements were performed only in *x* and *y* direction, so only estimated values for coefficients (PIR_{zz}, PVR_{zz}) in *z* direction (perpendicular to fins) are used.

Heat sinks in STAR-CCM+ are defined as “Volumetric heat sources” and mass sinks as “Species sources” in evaporator region. In order to speed up the whole process of CFD simulation set-up, multiple Java/C scripts/macros are developed. Scripts create basic evaporator geometry (block $L_x \times L_y \times L_{ztot}$), tube bundle (based on positions of tube section from MATLAB), subtraction of basic HE block and tube bundle to create evaporator region and set-up initial and inlet conditions, assign heat/mass sinks into evaporator region and heat/mass fluxes on tube surfaces. Simulation set-up is divided into 3 steps.

In order to set-up CFD model correctly, options for volume mass and heat sinks needs to be allowed. This requires using “Segregated Species” and “Segregated Fluid Temperature” solvers to be turned on, and 2 “Gas Component” to be specified. Using model of “Ideal Gas” is recommended, because no additional corrections are required as for example

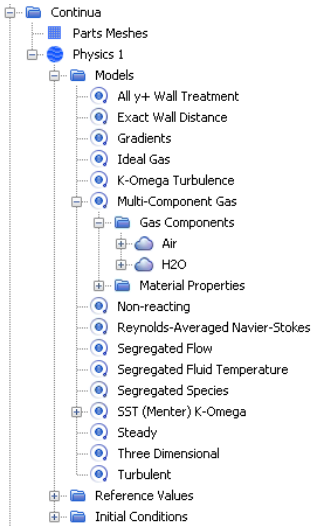


Fig. 60 Physics model required

4.1 Step 1 – Heat exchanger definition

Basic heat exchanger geometry is created by running script “HE_model_creation.java”. Running the script displays small menu, where a user is prompted to specify HE dimensions. (in mm) Dimensions must agree with dimensions specified in MATLAB simulation, otherwise, results will be pointless. After dimensions are specified, hit “SET-UP” button, the message in “Output” windows will display:

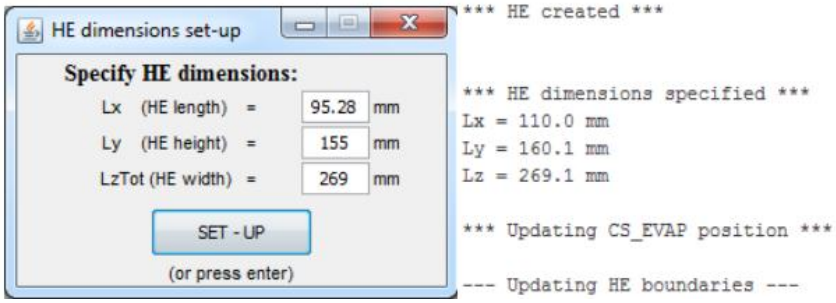


Fig. 61 HE definition message

New “3D-CAD Model” is created “HE_model” and then automatically transformed into “Part”.

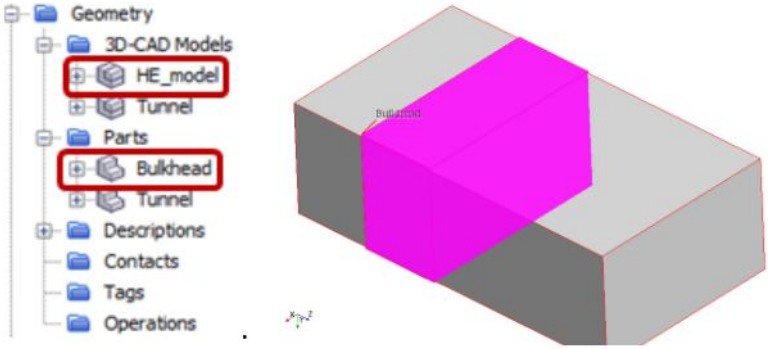


Fig. 62 New CAD model and part

CAD model is automatically created in origin of global CS. If any translation or rotation is required to fit for example inlet duct geometry, script “HE_transform_menu.java” with simple user interface should be used to perform these transformations.

4.2 Step 2 – Tube bundle definition

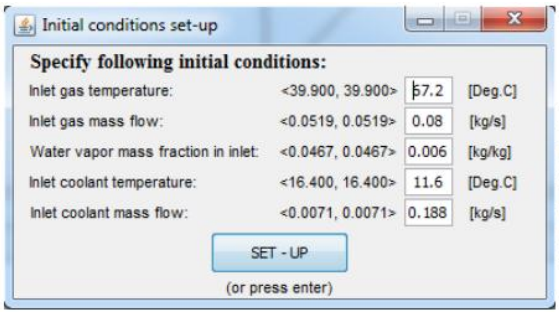
Only basic geometry is generated in the previous step (block with dimensions $L_x \times L_y \times L_{ztot}$) Next, tube bundle needs to be generated and then subtracted. By running script "Tube_create.java", all necessary parameters (pitch in *x* and *y* direction, number of tubes per row, number of tube rows, outer tube diameter) are loaded from *.csv files and tube bundle is created, following message displays in "Output" windows. All surfaces of tubes are automatically assigned into "Wall" boundary type, heat and mass flux through specific tube surface solved in MATLAB is also associated with this surface. Meshing operation is also modified – higher mesh density is forced on tube surface by the script.

```
*** Creating source tube (OD = 9.520000000000001mm) ***
No coincident faces or edges found.
*** Splitting tube surface ***
Part surface "Default" was split into 3 part surfaces.
*** Tranforming source tube to CS_HE coordinate system ***
*** Generating tube boundle ***
    Number of tube rows:    12
    Number of tube per row: 2
*** Deleting source tube ***
*** Creating subtract operation ***
*** Changing HE mesh target part ***
*** Changing HE region part ***
*** Creating Heat flux functions ***
*** Reading HF matrix file (TubeHF.csv) ***
*** Creating Mass flux functions ***
*** Reading HF matrix file (TubeMF.csv) ***
*** Setting-up tube boundaries (heat flux) ***
Tessellating CAD surface using medium density.
CAD based Boolean Subtract operation complete.
```

Fig. 63 Tube bundle creation messages

4.3 Step 3 – Inlet conditions specification

Script "UDF_create.java" loads information about inlet conditions and define all functions necessary to interpolate heat/mass sinks. A user is prompted to specify inlet conditions (range of inlet parameters is loaded automatically from MATLAB simulation file "VarParRange.csv"). If POD method is used, range of inlet parameters available is displayed in "< >", otherwise only the same inlet conditions as in MATLAB simulation should be used.




```
------------ Loading ranges for parameters -----------------
*** RANGE FILE FOUND! ***
Tg_inlet range: <39.90000000,39.90000000>
Mtg_inlet range: <0.05190000,0.05190000>
Wv_inlet range: <0.04671115,0.04671115>
Tc_inlet range: <16.40000000,16.40000000>
Mtc_inlet range: <0.00718000,0.00718000>
------------------------------------------------------------
```

Fig. 64 Initial conditions prompt and range of parameters loaded from MATLAB file

5 Summary and discussion

The main subject of this master's thesis is the heat and mass transfer modeling inside heat exchangers used in air dehumidification applications. The originality of the work resides in the development of a MATLAB model used to treat the heat and mass transfer in presence of humidity as source terms that are then imported in a CFD model commercial code to take into account the 3D complexity of the flow through the heat exchanger.

The focus of the work is specifically on the evaporator. The evaporator is located in the heat pump circuit along with a condenser, compressor and a capillary tube/reduction valve. There exist many types of heat exchanger geometries; this work is addressing the compact type (finned tube). Such heat exchanger is composed of a long tube that is bent to create multiple passes and a set of parallel fins are mounted on the tube bundle to increase the surface of exchange between circulating air and the heat exchanger. The fin corrugation is eventually taken into account in the final model by increasing the fin area by playing on some geometrical factors. A typical refrigerant used in heat pump circuits is R134a, which requires precise knowledge of the fluid properties in order to precisely simulate its state. General functions of pressure and temperature are created for all necessary fluid parameters, based on information from literature and National Institute of Standards and Technology.

Standard methods for heat exchanger analysis are covered in the first chapter. Well-known analytical methods - LMTD and ε-NTU and their limitations are described. Multiple ad-hoc numerical models from literature for similar applications are introduced and analyzed to evaluate which approach is the most suitable. An important statement is retrieved from reference [20], where the author analyzes the dependency of approximation error on a number of POD modes and a number of variable parameters. Reasonable precision is achieved by using three modes, but dependency on a range of variable parameters needs also to be considered.

Requirements on the mathematical model and its properties are introduced in the second chapter. The complexity of the model, mainly its dimensions, are selected to balance the precision of results and computational load. The MATLAB software is selected as a model development environment, because of its user-friendly environment, the relatively simple implementation of matrix operations and the available optimization tools for model validation and correction. The model itself is composed of multiple sub-procedures that solve various heat and mass transfer problems – fin surface temperature, condensate film temperature, local humid air parameters and conditions for condensation as well as the refrigerant temperature distribution along the tube bundle. The differential equations for heat and mass balance used in the model are then described in Chapter 3.7 and are solved with explicit first order Euler method. As a possible improvement of MATLAB model, additional code optimization and generalization can be mentioned: the current computation time is approximately 2 second per iteration for reasonable mesh density. Instead of using a finite difference method for the air temperature distribution and a finite element method for surface temperature distribution, it is preferable to use a 2D finite volume method, which may lead to further decrease in the computation time.

The model validation is covered in chapter 3.10. Validation itself is divided into 3 stages, to test all necessary components of MATLAB model and maximize the precision of results.

The general approach to model validation is developed, so it can be reproduced in the future with different model geometry. General optimization tool from MATLAB *Optimization Toolbox* is used – *fminsearch*, to identify effectiveness coefficients used in heat and mass transfer coefficient corrections. A dedicated test rig was built within the department as a parallel activity in order to obtain a set of experimental data to support the validation of the numerical approach. The first stage of the model validation was done with water as a coolant and no condensation on air side: the average error on the output parameters (air and coolant outlet temperature) was 2.6 % over 20 measured points which demonstrates a very good agreement. The second evaporator with different geometry was then used over 5 measurement points and confirmed the low error percentage. It was therefore concluded that the model has been properly designed and the coefficients for heat and mass transfer were correctly identified for the considered geometry. With the second stage of validation is the condensation on the air side added and the R134a is used as a refrigerant. Another 44 points were measured for the purpose of validating the mass transfer coefficient and the Sherwood mass transfer analogy. A good precision on the heat duty and the condensate mass flow rate as outcomes of the model was again achieved with a relative error to the experimental data set that is not exceeding 4.6%. Finally, the third stage of the model validation was focusing on the precision of the data transfer between MATLAB and Starccm+, a CFD general purpose commercial package. Great agreements between model outputs and experimental data were once more achieved. A user manual to MATLAB model is included in this thesis, describing the setup inlet conditions, evaporator geometry, solver parameters and control solution stability.

The final part of the thesis is dedicated to creating an automatic procedure of data transfer. This procedure is designed to transform output fields from MATLAB simulation, generate *csv files, import these files into Starccm+ CFD code and set-up the whole simulation. The results from CFD simulation are compared to results from MATLAB model for various inlet conditions and geometrical parameters of the evaporator. The agreement with experimental data was analyzed in this step of the validation and overall differences in outlet parameters, total heat duty or condensate mass flow are not exceeding 3.2%.

As a continuation of this work, the model of the condenser should be created. It can be simply derived from evaporator model, by excluding condensation procedures and other simplifications, so further time reduction of computation time is expected. To complete the whole heat pump circuit, simplified model of a compressor and reduction valve should be also developed, to be able to simulate circuit as a unit, without measurement of particular parameters, like refrigerant temperature in condenser outlet and evaporator inlet, etc. This would also require to run the model as a transient analysis, but probably also as quasi-steady state to simulate evolution from an initial (stagnation) point. The input parameters could be only initial pressure in a pipe system, refrigerant charge and compressor speed. This ultimate goal would provide a great tool for optimization of all components of heat pump circuit with the scope to reduce component costs but also allow to test system control procedures.

List of figures

List of tables

References

[1] SMARTCLIMA. *Finned U Tube Air Heat exchanger* [online] [29.6.2017] Available from: http://www.smartclima.com/finned-u-tube-air-cooled-heat-exchanger.htm

[2] NUSSELT, W. *Die Oberflächenkondensation des Wasserdampfes*. Zeitschrift des Vereines Deutscher Ingenieure 60(27), 541-546 (1916).

[3] BROMLEY, L.R.A. *Heat transfer in condensation – effect of heat capacity of condensate*. Industrial and Engineering Chemistry 44 (1952) 2966 - 2969.

[4] ROHSENOW, W.J. *Condensation – Handbook of heat transfer*. McGraw-Hill. ISBN: 0-07-053555-8 Available from: http://160592857366.free.fr/joe/ebooks/Mechanical%20Engineering%20Books%20Collection/HEAT%20TRANSFER/handbook%20of%20HeatTransfer.pdf

[5] MINKOWYCZ, W.J; SPARROW, E.M. Condensation heat transfer in the presence of a non-condensables, interfacial resistance, superheating, variable properties and diffusion. *International Journal of Heat and Mass Transfer* [online]. Science Direct. February 2003, 9 (2003) 1125-1144. ISSN: 0017-9310. [17 December 2016]. Available from: www.elsevier.com/locate/ijhmt

[6] LEBEDEV, P.D.; BAKLASTOV, A.M.; SERGAZIN, Z.F. Aerodynamics heat and mass transfer in vapor condensation from humid air on a flat plate in longitudinal flow in asymmetrically cooled slot. *International Journal of Heat and Mass Transfer* [online]. Science Direct. February 1969, 12 (1969) 833-841. ISSN: 0017-9310. [17 December 2016]. Available from: www.elsevier.com/locate/ijhmt

[7] MAMYODA, T.; ASANO, K. Experimental study of condensation of vapors in the presence of non-condensable gas on a short horizontal tube. *Journal of Chemical Engineering of Japan* [online]. August 1944, 27 (1994) 485-490. ISSN: 0021-9592. [17 December 2016]. Available from: http://www.scej.org/jcej/

[8] COLBURN, A.P.; HOUGEN, O.A. Experimental study of condensation of vapors in the presence of non-condensable gas on a short horizontal tube. *Industrial & Engineering Chemistry* [online]. November 1934, 26 (1934) 1178-1182. ISSN: 1520-5045. [17 December 2016]. Available from: http://pubs.acs.org/doi/pdf/10.1021/ie50299a011

[9] CHE, D.; DA, Y.; ZHUANG, Z. Heat and mass transfer characteristics of simulated high moisture flue gases. [online]. Web of Science. January 2005, 41 (2005) 250-256. ISSN: 0947-7411. [17 December 2016]. Available from: www.elsevier.com/locate/ijhmt

[10] GROFF, M.K.; ORMISTON, S.J. Numerical solution of film condensation from turbulent flow of vapor-gas mixture in vertical tubes. *International Journal of Heat and Mass Transfer* [online]. Science Direct. September 2007, 50 (2007) 3899-3912. ISSN: 0017-9310. [17 December 2016]. Available from: www.elsevier.com/locate/ijhmt

[11] PATANKAR, S.V. Numerical Heat Transfer and Fluid Flow. Hemisphere Publishing Corporation, 1980. ISBN: 0-07-048740-5. Online version available from: http://www.ewp.rpi.edu/hartford/~ernesto/F2012/CFD/Readings

[12] PAVLŮ, Jaroslav. *Vývoj výpočetního modelu a metodiky pro výpočet kondenzátorů s minikanálky* [online]. Vysoké učení technické v Brně. Fakulta strojního inženýrství, 2012 [29.6.2017]. Available from: http://hdl.handle.net/11012/5951

[13] INCROPERA, F. P. *Fundamentals of heat and mass transfer*. 7th ed. USA: John Wiley& Sons, Inc., 2011. ISBN 13 978-0470-50197-9

[14] BOWMAN, R.A.; MUELLER, A.C.; NAGLE, W.M. *Mean Temperature difference in Design*. Transaction of A.S.M.E. [29.6.2017]

[15] WARD, JOHN A. *Mean temperature difference* [online] [29.6.2017] Available from: http://www.thermopedia.com/content/945/

[16] JUN-DE, Li; MOHAMMAD, Saraireh GRAHAM Thorpe. Condensation of vapor in the presence of non-condensable gas in condensers. *International Journal of Heat and Mass Transfer* [online]. Elsevier. May 2011, 54 (2011) 4078–4089. ISSN: 0017-9310. [14 April 2015]. Available from: http://www.elsevier.com/locate/ijhmt

[17] RUIVO, C.R.; DOMINGUEZ-MONUZ, F.; COSTA, J.J. Simplified model of finned-tube heat exchangers based on the effectiveness method and calibrated with manufacturer experimental data. *Applied Thermal Engineering* [online]. Elsevier. September 2016, 111 (2017) 340 - 352. ISSN: 1359-4311. [17 December 2016]. Available from: www.journals.elsevier.com/applied-thermal-engineering

[18] WAHABI, Y.; GHORAB, M.; ENTCHEV, E. 3D CFD study of the effect of inlet air flow maldistribution on plate-fin-tube heat exchanger design and thermal-hydraulic performance. *International Journal of Heat and Mass Transfer* [online]. Science Direct. June 2016, 101 (2016) 527–541. ISSN: 0017-9310. [17 December 2016]. Available from: www.elsevier.com/locate/ijhmt

[19] CLÁUDIO, Melo; CHRISTIAN, J.L. Hermes. A heat transfer correlation for natural draft wire-and-tube condensers. *International journal of refrigeration* [online]. Elsevier. June 2008, 32 (2009) 546–555. SSN: 0140-7007. [14 April 2015]. Available from: http://www.elsevier.com/locate/ijrefrig

[20] ADAMEC, J. *Modelování přestupu tepla a hmoty ve filmové výplni.* Praha: ČVUT 2014. Diplomová práce (Ing.), ČVUT, Fakulta strojní, Ústav mechaniky tekutin a termodynamiky.

[21] BOROVIČKA, M. Mathematical model of evaporator for heat pump circuits. *Student's conference STC* [online]. April 2016, ISBN: 978-80-01-05929-6. [17 December 2016]. Available from: stc.fs.cvut.cz/history/2016/sbornik/index-en.html

[22] NOŽIČKA, J. *Základy termomechaniky*. Praha: ČVUT, 2008. ISBN 978-80-01-04022-5

[23] Verein Deutscher Ingenieure. *VDI-Wärmeatlas*. 7th ed. Germany - Düsseldorf: VDI-Verlag, 1994. ISBN 3184013618, 9783184013615

[24] AALBORG UNIVERSITET. *Finite Element Method – 2D heat conduction* [online] [20.8.2016] Available from: http://www.wind.civil.aau.dk/lecture/7sem_finite_element/lecture_notes/Lecture_3_4.pdf

[25] KLOPPERS, J.C.; Kröger, D.G. The Lewis factor and its influence on the performance prediction of wet-cooling towers. *International Journal of Thermal Science.* [online].

Science Direct. April 2005, 44 (2005) 879-884. ISSN: 1290-0729. [3 March 2016]. Available from: www.journals.elsevier.com/ijts

[26] BOURILLOT, C. TEFERI, numerical model for calculating the performance of an evaporative cooling tower. *EPRI Report* CS – 3212 – SR, August 1983. [20 October 2016].

[27] LEWIS, W.K. A. The evaporation of a liquid into a gas. *International Journal of Heat and Mass Transfer* [online]. Elsevier. February 1952, 5 (1952) 109-112. ISSN: 0017-9310. [20 October 2016]. Available from: www.elsevier.com/ locate/ijhmt

[28] ŠESTÁK, J.; RIEGER, F. *Přenos hybnosti, tepla a hmoty*. Praha: ČVUT, 2005. ISBN 80-01-02933-6

[29] NOŽIČKA, J. *Mechanika tekutin*. Praha: ČVUT, 2006. ISBN 80-01-02865-8

[30] GROFF, M.K.; ORMISTION, S.J.; SOLIMAN, H.M. Numerical solution of film condensation from turbulent flow of vapor-gas mixtures in vertical tubes. *International Journal of Heat and Mass Transfer.* [online]. Science Direct. April 2007, 50 (2007) 3899–3912. ISSN: 0017-9310. [3 March 2016]. Available from: www.journals.elsevier.com/ijhmt

[31] CHENG, Li; JUNMING, Li. Laminar forced convection heat and mass transfer of humid air across a vertical plate with condensation. *Chinese Journal of Chemical Engineering.* [online]. Science Direct. December 2011, 19 (6) 944–954 (2011). ISSN: 1004-9541. [17 December 2016]. Available from: www.journals.elsevier.com/chinese-journal-of-chemical-engineering

[32] VOLCHKOV, E.P.; TEREKHOV, V. Numerical study of boundary-layer heat and mass transfer in a forced flow of humid air with surface steam condensation. *International Journal of Heat and Mass Transfer.* [online]. Science Direct. March 2004, 47 (2004) 1473-1481. ISSN: 0017-9310. [3 March 2016]. Available from: www.journals.elsevier.com/ijhmt

[33] The International Association for the Properties of Water and Steam. *Properties of Ordinary Water Substance for General and Scientific Use*. Doorweth: IAPWS, 2009.

[34] POOYA, M.; CHAO, Z. Three-dimensional numerical model for the two-phase flow and heat transfer in condensers. *International Journal of Heat and Mass Transfer* [online]. Elsevier. November 2014, 81 (2015) 618–637. ISSN: 0017-9310. [14 April 2015]. Available from: http://www.elsevier.com/locate/ijhmt

[35] JUSSI, S.; JUHA, K.; ESA, V.; SAMULI, S. Comparison of power plant steam condenser heat transfer models for on-line condition monitoring. *Applied Thermal Engineering* [online]. Elsevier. September 2013, 62 (2014) 37-47. ISSN: 1359-4311. [14 April 2015]. Available from: http://www.elsevier.com/locate/apthermeng

[36] J.S, Hu; CHRISTOPHER, Y.H. Chao. An experimental study of the fluid flow and heat transfer characteristics in micro-condensers with slug-bubbly flow. *International journal of refrigeration* [online]. Elsevier. May 2007, 30 (2007) 1309-1318. ISSN: 0140-7007. [14 April 2015]. Available from: http://www.elsevier.com/locate/ijrefrig

[37] NYERS, Jozsef GARBAI, Laszlo NYERS, Arpad. A modified mathematical model of heat pump's condenser for analytical optimization. *Energy* [online]. Elsevier. January 2015, 80 (2015) 706e714. SSN: 0360-5442. [14 April 2015]. Available from: http://www.elsevier.com/locate/energy

[38] ARSENYEVA, Olga TOVAZHNYANSKY, Leonid KAPUSTENKO, Petro PEREVERTAYLENKO, Oleksander KHAVIN, Gennadiy. Investigation of the new corrugation pattern for low pressure plate condensers. *Applied Thermal Engineering* [online]. Elsevier. February 2011, 31 (2011) 2146-2152. ISSN: 1359-4311. [14 April 2015]. Available from: http://www.elsevier.com/locate/apthermeng

[39] TSILINGIRIS, P.T. Thermophysical and transport properties of humid air at temperature range between 0 and 100℃. *Energy conversion and managemant* [online]. Elsevier. November 2007, 49 (2008) 1098-1100. ISSN: 0196-8904. [14 April 2015]. Available from: http://www.elsevier.com/locate/enconman

[40] REZK, K.; FORSBERG, J. Geometry development of the internal duct system of a heat pump tumble dryer based on fluid mechanic parameters from a CFD software. *Applied Energy* [online]. Elsevier. December 2010, 88 (2011) 1596–1605. ISSN: 0306-2619. [14 April 2015]. Available from: www.elsevier.com/ locate/apenergy

[41] AC HACKER. E*corenovator - forum* [online]. © EcoRenovator. Available from: http://ecorenovator.org/forum/18881-post1066.html

[42] WATANABE, N.; MASANORI, A. Correlative relationship between geometric arrangement of drops in dropwise condensation and heat transfer coefficient. *International Journal of Heat and Mass Transfer* [online]. Elsevier. October 2016, 105 (2017) 597–609. ISSN: 0017-9310. [20 October 2016]. Available from: www.elsevier.com/ locate/ijhmt

[43] HUZAYYIN. A.S.; NADA, S.A.; ELATTAR, H.F. Air-side performance of wavy finned tube direct expansion cooling and dehumidifying air coil. *International journal of refrigeration* [online]. Science Direct. October 2006, 30 (2007) 230–244. ISSN: 0140-7007. [17 December 2016]. Available from: www.elsevier.com/ locate/ijrefrig

Appendices

Appendix 1 – Working folder description

There are 9 folders in working (root) directory of MATLAB model and 4 main *.m files. Folders and file should not be deleted in order to maintain all functionalities of the model.

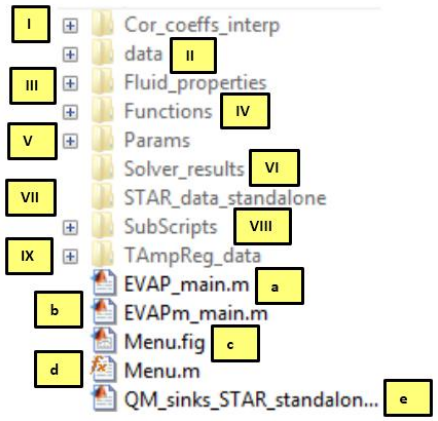


Fig. 65 Model folder content

Folder content:

I. "Cor_coeffs_interp" folder – contains interpolation function
II. "data" fodler – contains Menu defaults
III. "Fluid_properties" folder – contains fluid properties of water and R134a
IV. "Functions" folder
V. "Params" folder – contains HE geometrical parameters, solver parameters, inlet conditions
VI. "Solver results" folder – contains data generated during simulation in sub-folder specified by user
VII. "STAR_data_standalone" folder – contains data with heat and mass sinks for STAR-CCM+ created separately after simulation.
VIII. "SubScripts" folder – contains model subroutines
IX. "TAmpReg_data" folder – contains regressions generated during model validation

a. "EVAP_main.m" file – main execution file for Text/*.m file interface (See chapter *Text/*.m file interface)*
b. "EVAPm_main.m" file – secondary execution file for Menu interface (this file is automatically run after pressing "Solve HE!" button in menu, do not run this file separately)
c. "Menu.fig" file – contains layout of user menu
d. "Menu.m" file – main execution file for user menu (See chapter *Menu interface*)
e. "QM_sinks_STAR_standalone.m" file – execution file for heat/mass sinks generation

Appendix 2 – File description

a) "*Evap_Params_standalone.m*" contains solver parameters, HE dimensions, tube bundle parameters and superposition parameters (see note 1 for more details)

b) "*Init_standalone.m*" contains inlet air conditions, inlet refrigerant conditions and refrigerant choice (see note 2 for more details)

c) "EVAP_main.m" main execution file that starts simulation with parameters specified in a) and b). (RMB →Run or F9)

This approach is probably more comfortable for advanced MATLAB users and allows you to directly control advanced solver parameters and more easily debug potential errors during simulation.

Heat exchanger and solver parameters

All geometrical parameters of heat exchanger and tube bundle is specified in first part of "Evap_Params_standalone.m" file – "BASIC USER INPUT" along with basic solver parameters, if "text" interface is used.

```
1  %% ######################################### Parameters ########################################
2
3  %% ################################# START OF BASIC USER INPUT \\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
4  %% ##########################################################################################
5
6  %% HE dimensions
7  s              = 0.12e-3;    % evap fin thickness [s] = m
8  Lx             = 87e-3;      % evap length [Lx] = m
9  Ly             = 150e-3;     % evap height [Ly] = m
10 Lz             = 2.0e-3 - s; % fin pitch [Lz] = m
11 deltaLx        = 21.65e-3;   % Tubes x pitch
12 deltaLy        = 25e-3;      % Tubes y pitch
13 Dtube_out      = 9.52e-3;    % Tube outer diameter
14 Tube_thickness = 0.7e-3;     % Tube wall thickness
15 Dtube_in       = Dtube_out - 2*Tube_thickness;
16
17 %% Calculation and discretization parameters
18 Telem = 37;    % number of elements for tube  37  81  97 249 321
19 KMit  = 100;     % number of iterations
20 Cit   = 30;    % Coolant temp iterations
21 Mix   = 2;     % Mixing
22 URFc  = .4;     % Under relaxation factor for coolant distribution solver
23 [N, M, ErrSurf] = M_elem(Telem, Lx, Ly, Dtube_out);
24
25 %% Tube bundle parameters
26 Nd    = 117;   % number os desks in condenser
27 nRows = 12;    % number of rows
28 nTPR  = 2;     % tubes per row
29
30 %% Other parameters
31 PlotResults = true;  % Plot results after computation
32
33 %% ################################# END OF BASIC USER INPUT \\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
34 %% ##########################################################################################
```

Line description:

7- evaporator fin thickness (sheet metal thickness) (in m)

8- evaporator length (in m)

9- evaporator height (in m)

10- fin pitch (in m)

11- tube pitch in x direction (in m)

12- tube pitch in y direction (in m)

13- outer tube diameter (in m)

14- tube wall thickness (in m)

18- *Telem* parameter – number of elements per tube cross-section

19- number of iterations for solution

20- number of iterations for refrigerant distribution solver

21- mixing parameter for temperature averaging

22- *URF* for refrigerant temperature solver

23- automatic solution for *N*, *M* (number of elements in x and y direction) based on minimization of area error of tube cross-section

26- number of fins (without end plates)

27- number of tube rows in tube bundle

28- number of tubes per row in tube bundle

Second part of file includes selection of specific solver (*STANDART*, *CAREFUL*, *FORCED*) and advanced parameters for these solvers.

```
55      %% ////////////////////////////////// START OF ADVANCED USER INPUT \\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
56      %% ##########################################################################################################
57
58      %% Under relaxation factor for heat duty + redistribution parameter
59 -    URFQmod   = 0.529687; URFQmodALL = URFQmod;    % Initial URFg
60 -    dURFq     = .0;       % Increment of redistribution factor
61 -    IIAllowed = false;   % Allow alg. to increase number of iterations if needs to be
62 -    URFqmodif = false;   % Allow alg. to modify URFq to achive better HB error
63 -    HBLIM     = .00;    % Limiting precision for increasing dURFq
64 -    dURFqLIM  = .00;    % Value for dURFq if error(HB) > HBLIM
65
66      %% Carefull solver parameters
67 -    CarefulI  = false;    % Slow convergence, but stability
68 -    RadaLim   = 350;    % Initial guess for SUPERHEATING point location
69 -    dRada     = 15;      % Increment for SUPERHEATING point location
70 -    dRadaLIM  = 4;      % Value for dRada if error(Qfr) > QfrLIM
71 -    QfrLIM    = .015;    % Limiting precision for increasing dRada
72 -    URFQstart = 20;     % Delay(#iterations) for redistribution factor modification
73
74      %% Constant ref.temp solver (FORCED solver)
75 -    CPForced  = false;   % Forces algorithm to keep ref.temp constant
76
77      %% Create STAR sources
78 -    STARs = false;
79
80      %% ////////////////////////////////// END OF ADVANCED USER INPUT \\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
81      %% ##########################################################################################################
```

Line description:

59- initial guess for redistribution correction parameter

60- initial increment for redistribution correction parameter

61- 61-automatic increase of number of iterations if *HB* error is not sufficient (Boolean)

62- 62-*RCS* option (Boolean)

63- minimal *HB* error (%/100)

64- increment for case minimal HB error is achieved (should be small ~(0.01-0.03)), just to avoid stagnation of *URFqMod(54).*

67- *CAREFUL* solver option (Boolean)

68- initial guess for superheating point

69- initial increment for superheating point (should be changed based on how close the simulation is getting close to convergence)

70- increment for case minimal error of Q_{fr} is achieved (should be small ~(2-5), just to avoid stagnation of superheating point)

71- minimal error of Q_{fr} parameter (%/100)

72- initial delay (number of iterations) for start of *RCS* solver (*URqstart(53) = 30* recommended, but can be increased up to KMit(12)/2)

75- *FORCED* solver options (Boolean)

78- heat and mass sinks for STAR-CCM+ will be created after simulation is complete (Boolean)

Inlet conditions specification

All inlet and boundary conditions are specified in “Init_standalone.m” file, if “text” interface is used.

```
1   %% ################################### Initial/Inlet conditions ################################
2
3   %% ////////////////////////////////// START OF USER INPUT \\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
4   %% ##########################################################################################
5
6   %% Initial/Inlet conditions
7   % % Evaporator
8   Tg0      = 39.9;  % [Tg] = Deg.Cels    note: uniform inlet temp. assumed
9   Tc0      = 16.4;  % [Tc] = Deg.Cels    note: uniform inlet temp. assumed
10  T_tubes0 = Tc0;   % [T_tubes] = Deg.Cels
11  Mtg0     = 32.2/1000;   %[mtg] = kg/s       note: uniform inlet velocity assumed
12  Mtc0     = 7.18/1000;   %[mtg] = kg/s       note: uniform inlet velocity assumed
13  wv0      = 0.04671;   % [wv] = kg(water vapor)/kg(moist air)      note: uniform inlet mass fraction assumed
14  pg0      = 99240;   % [pa] = Pa
15  pc0      = 501000;   % [pc] = Pa
16
17  % Condenser (only for R134a)
18  TCondOut = 55.4;    % Outlet condenser temperature; [TCondOut] = Deg.Cels
19  pCond    = 1584000; % Pressure in condenser [pCond] = Pa
20
21  %% Coolant/refrigerant + correction for pressure
22  CName    = 'R134a'; % 'WATER' or 'R134a';
23  PressCor = true;    % 1 = derives correct pressure from inlet temperature( = boiling temp.)(only for R134a)
24  %% ////////////////////////////////// END OF USER INPUT \\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
25  %% ##########################################################################################
```

Line description:

8- Inlet air temperature (in ℃)

9- Inlet refrigerant temperature (in ℃)

10- Initial guess for outer tube surface temperature (automatically set to T_{c0})

11- Inlet mass flow of air (in kg/s)

12- Inlet mass flow of refrigerant (in kg/s)

13- Inlet humidity (kg of water vapor per kg of moist air)

14- Inlet air pressure (barometric pressure) (in Pa)

15- refrigerant pressure in evaporator (only for R134a) (in Pa)

18- condenser outlet refrigerant temperature (only for R134a) (in ℃)

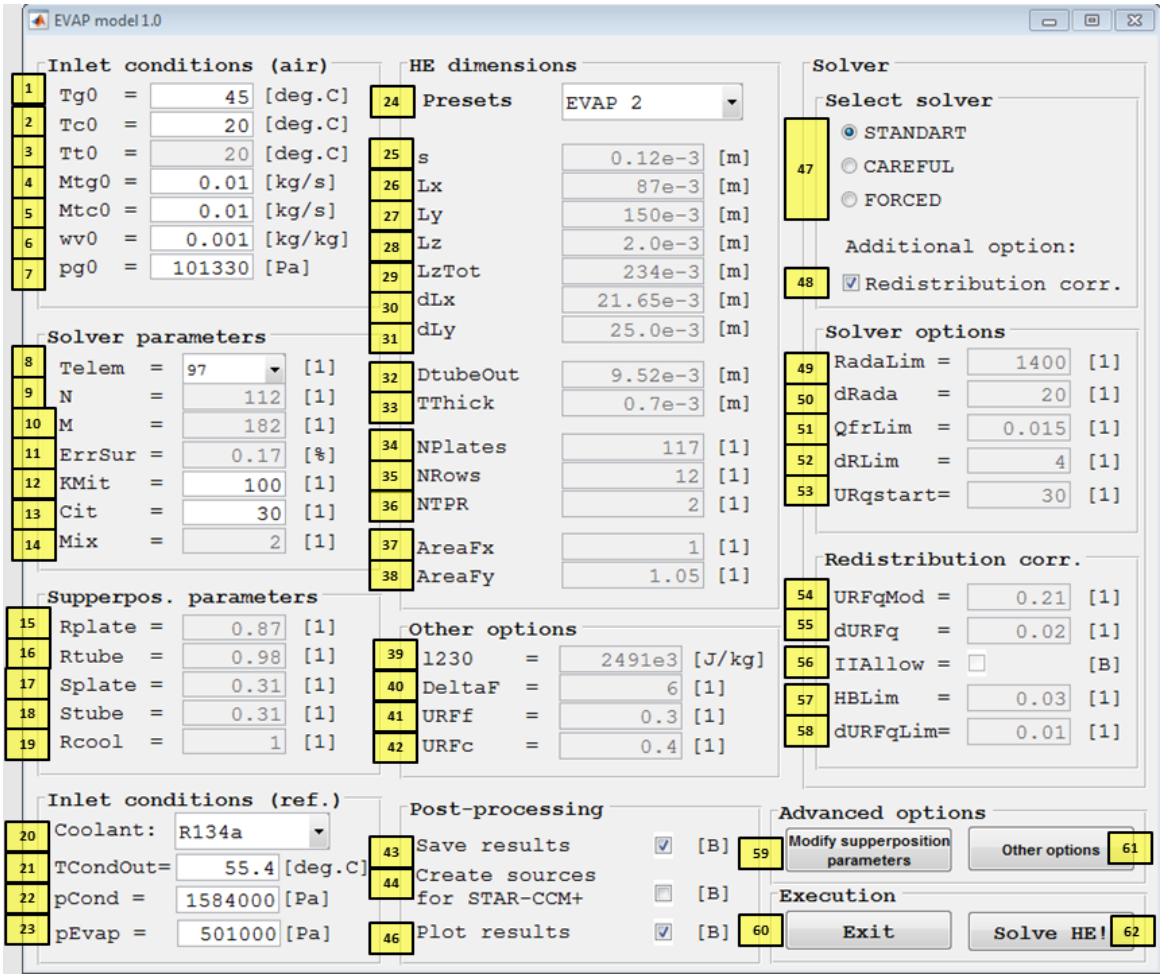
19- refrigerant pressure in condenser (only for R134a) (in Pa)

22- refrigerant choice

23- pressure correction (pressure modified according to T_{c0})

Appendix 3 – Menu interface

User menu for version 1.0 of evaporator mathematical model is shown in picture bellow.



Menu item description:

Inlet conditions (air)

1- Inlet air temperature (in ℃)

2- Inlet refrigerant temperature (in ℃)

3- Initial guess for outer tube surface temperature (automatically set to T_{c0})

4- Inlet mass flow of air (in kg/s)

5- Inlet mass flow of refrigerant (in kg/s)

6- Inlet humidity (kg of water vapor per kg of moist air)

7- Inlet air pressure (barometric pressure) (in Pa)

Solver parameters

8- Number of elements in tube section

9- number of elements in x direction (automatically solved from *Telem*(8)-by minimization of *ErrSur*(11))

10- number of elements in y direction (automatically solved from *Telem*(8)-by minimization of *ErrSur*(11))

11- error of tube section area (%)

12- number of iterations (may be increased automatically if *IIAllow(56)* = *true*)

13- number of inner iterations for refrigerant temperature distribution solver (*Cit*(13*)* = *30* recommended)

14- parameter for temperature averaging (should not be changed)

Superposition parameters (modifiable only if "Modify sup. parameters" button (59) is pressed)

15- correction for effective HT coefficient on fins (should not be changed, calibrated from experiments)

16- correction for effective HT coefficient on tubes (should not be changed, calibrated from experiments)

17- correction for effective MT coefficient of fins (should not be changed, calibrated from experiments)

18- correction for effective HT coefficient of tubes (should not be changed, calibrated from experiments)

19- correction for effective HT coefficient inside the tube (not calibrated, assumed to be = 1)

Inlet conditions (refrigerant)

20- refrigerant choice – *WATER/R134a*

21- condenser outlet refrigerant temperature (only for R134a) (in ℃)

22- refrigerant pressure in condenser (only for R134a) (in Pa)

23- refrigerant pressure in evaporator (only for R134a) (in Pa)

HE dimensions

24- presets for heat exchanger dimensions

25- thickness of fin (in m)

26- EVAP length (in m)

27- EVAP height (in m)

28- fin pitch (in m)

29- EVAP width (in m)

30- tube pitch in x direction (in m)

31- tube pitch in y direction (in m)

32- tube outer diameter (in m)

33- thickness of tube wall (in m)

34- number of fins (without end plates)

35- number of tube rows (#)

36- number of tubes per row (#)

37- effective area factor in x direction

38- effective area factor in y direction

Other options (modifiable only if “Other options” button (61) is pressed)

39- latent heat of evaporation at 0 ℃ (in J/kg) – should not be changed

40- initial guess for film thickness ratio on tubes

41- URF for fin temperature distribution (*URFf(41) = 0.3* recommended, may be changed if stability is not violated to achieve faster convergence)

42- URF for refrigerant temperature distribution (*URFc(42) = 0.4* recommended, may be changed if stability is not violated to achieve faster convergence)

Post processing

43- results will be saved after simulation is complete (Boolean)

44- heat and mass sinks for STAR-CCM+ will be created after simulation is complete (Boolean)

45- plot results after simulation is complete (Boolean)

Select solver

47- solver choice

48- correction for heat flow redistribution

Solver options (modifiable only if CAREFUL solver (47) is chosen)

49- initial guess for superheating point

50- initial increment for superheating point (should be changed based on how close the simulation is getting close to convergence)

51- minimal error of Q_{fr} parameter (%/100)

52- increment for case minimal error of Q_{fr} is achieved (should be small $\sim(2-5)$, just to avoid stagnation of superheating point)

53- initial delay (number of iterations) for start of “Redistribution correction” solver (*URqstart(53) = 30* recommended, but can be increased up to *KMit(12)/2*)

Redistribution correction

54- initial guess for redistribution correction parameter

55- initial increment for redistribution correction parameter

56- automatic increase of number of iterations if HB error is not sufficient (Boolean)

57- minimal HB error (%/100)

58- increment for case minimal HB error is achieved (should be small $\sim(0.01-0.03)$), just to avoid stagnation of *URFqMod(54)*

Advanced options

59- modify superposition parameters (Boolean) - allow modification of lines 15, 16, 17, 18, 19.

61- other options (Boolean) – allow modification of lines 39, 40, 41, 42.

Execution

60- proper exit from application (using  is not recommended)

62- solver simulation is initiated (simulation can be terminated by CTRL+C anytime)

Appendix 4 - Basic report and enthalpy report

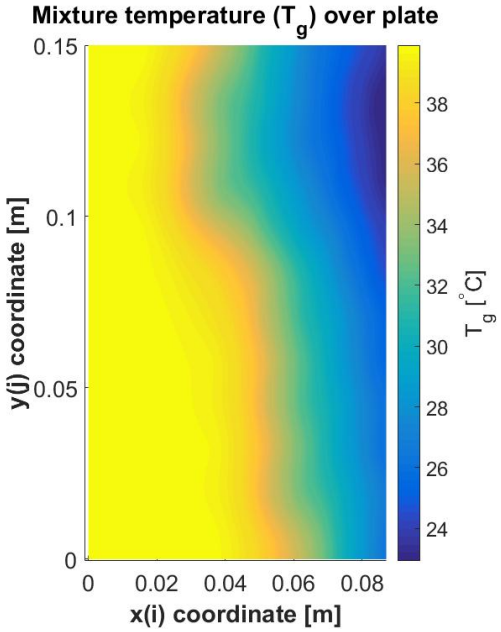
```
------------- Basic report -------------
Elements in x(i) direction: 65
Elements in y(j) direction: 112
Number of condenser plates: 117
Total number of elements:   7104
Total number of iterations: 14560000
Total time to compute:      248.8122 (s)

INLET PARAMETERS:
Mixture mass flow:          0.032200  [kg/s]
Coolant mass flow:          0.007180  [kg/s]
Mean mixture temperature:   39.90000  [deg.C]
Mean coolant temperature:   16.40000  [deg.C]
Mean mass fraction of w.v:  0.046710  [1]
Mean relative humidity:     1.000000  [1]
Mean specific humidity:     0.048999  [1]

OUTLET PARAMETERS:
Mixture mass flow:          0.031853  [kg/s]
Coolant mass flow:          0.007180  [kg/s]
Film mass flow:             0.000359  [kg/s]
Mean mixture temperature:   34.73338  [deg.C]
Mean coolant temperature:   39.86010  [deg.C]
Mean film temperature:      30.90725  [deg.C]
Mean mass fraction of w.v:  0.036322  [1]
Mean relative humidity:     1.000000  [1]
Mean specific humidity:     0.037691  [1]
-------------------------------------------
```

```
------------- Enthalpy report -------------
All PLATES CONDITIONS:
INLET PARAMETERS:
Dry air enthalpy flow:     1231.55730 (J/s)
Water vapor enthalpy flow: 3858.6892 (J/s)
Coolant enthalpy flow:     114.30107 (J/s)
Overall enthalpy flow:     5090.2465 (J/s)

OUTLET PARAMETERS:
Dry air enthalpy flow:     1070.90092 (J/s)
Water vapor enthalpy flow: 2957.78412 (J/s)
Coolant enthalpy flow:     1189.2489 (J/s)
Cond. film enthalpy flow:  42.184829 (J/s)
Overall enthalpy flow:     5145.1837 (J/s)
Heat transf. from G to C:  1074.3138 (J/s)
L/S heat ratio - air:      0.8069     (1)
L/S heat ratio - ref.:     0.8536     (1)

Error of discretization (AS):   1.079264 (%)
Error of HB:                    -0.059018 (%)
-----------------------------------------------
```

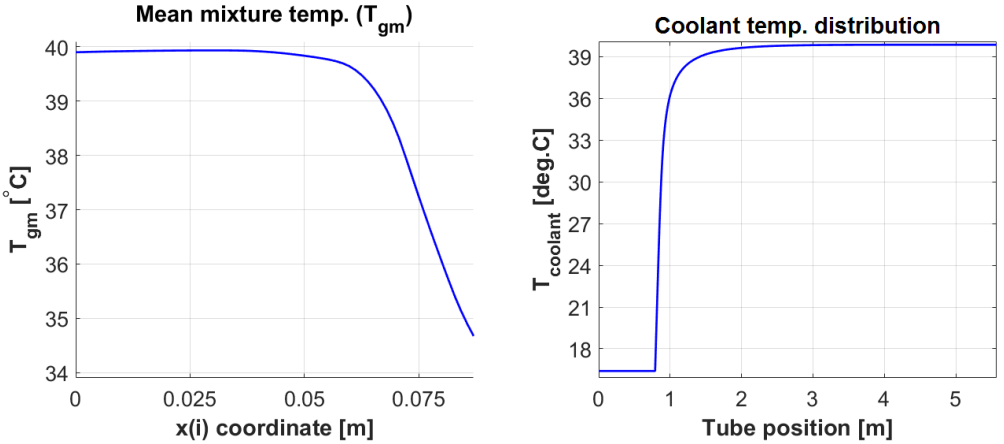
Appendix 5 - HT report and sinks creation

```
------------------- HT report -------------------
Mean HTC on plates:       34.62  [W.m^-2.K^-1]
Mean HTC on tubes:        3.20  [W.m^-2.K^-1]
Mean HTC overall:         24.93  [W.m^-2.K^-1]

Mean temp. difference (plates):    1.72  [Deg.C]
Mean temp. difference (tubes):     3.24  [Deg.C]
LMTD (Log.mean temp. difference):  2.98  [Deg.C]

Outer HT area (plates):   2.6313  [m^2]
Outer HT area (tubes):    0.1565  [m^2]
Outer HT area (total):    2.7878  [m^2]
Inner HT area (total):    0.1665  [m^2]
-------------------------------------------------

---- Sources for STAR-CCM+ not created! ----
```

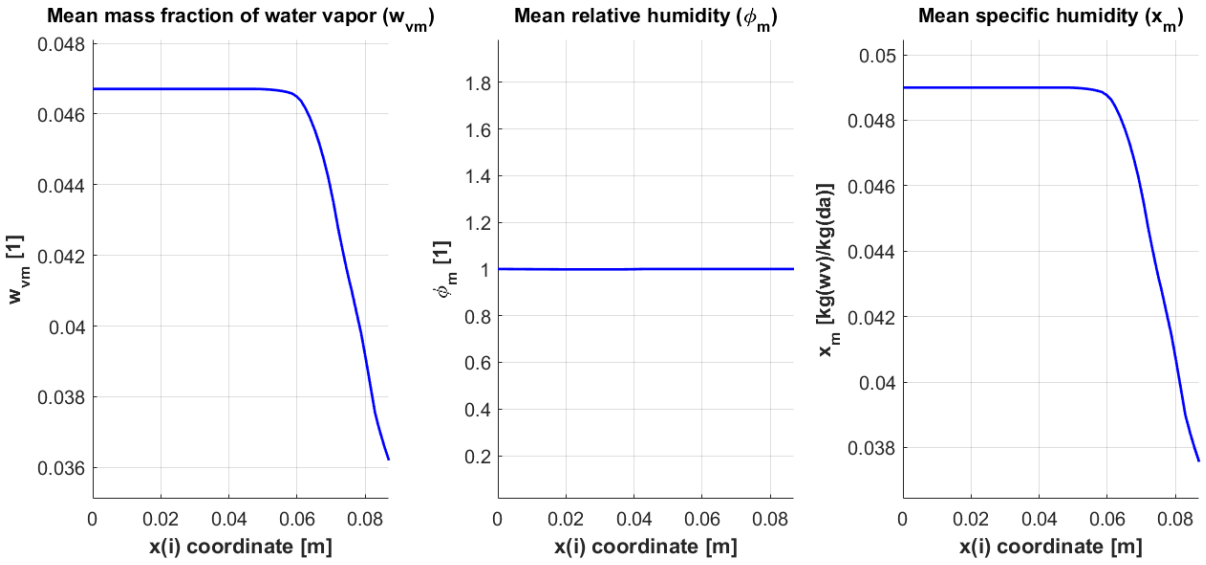
Appendix 6 – Mixture temperature plot



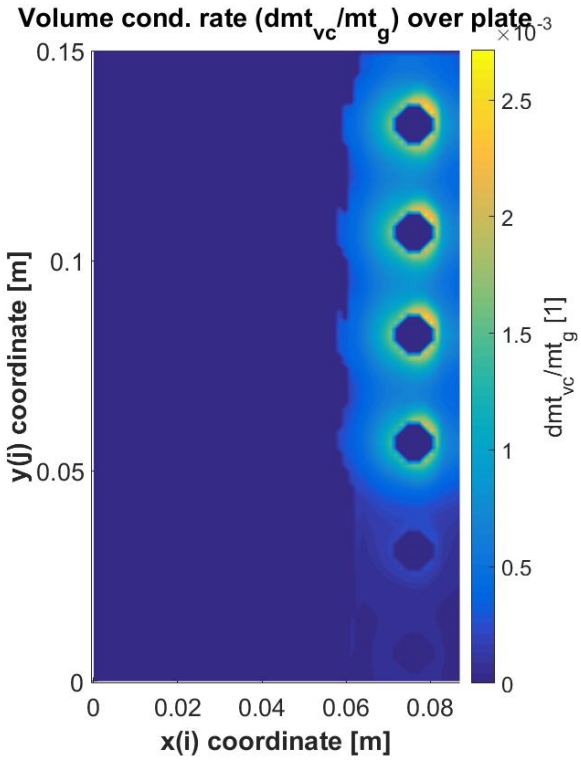
Appendix 7 – Mean temperature plot



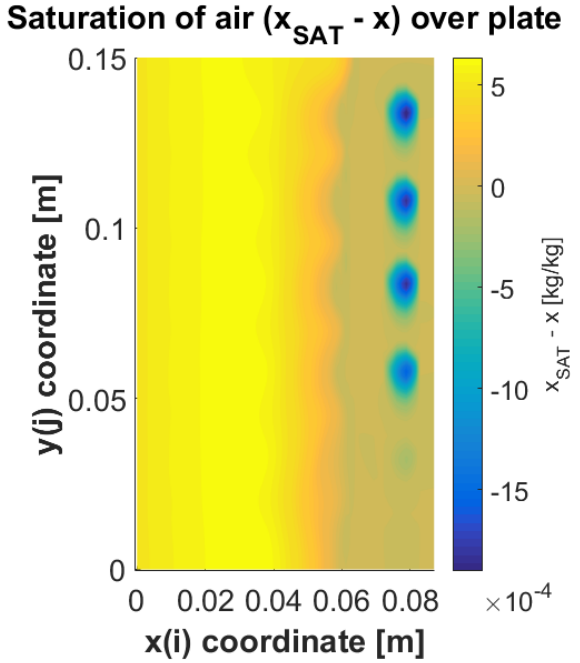
Appendix 8 – Mean humidity plot



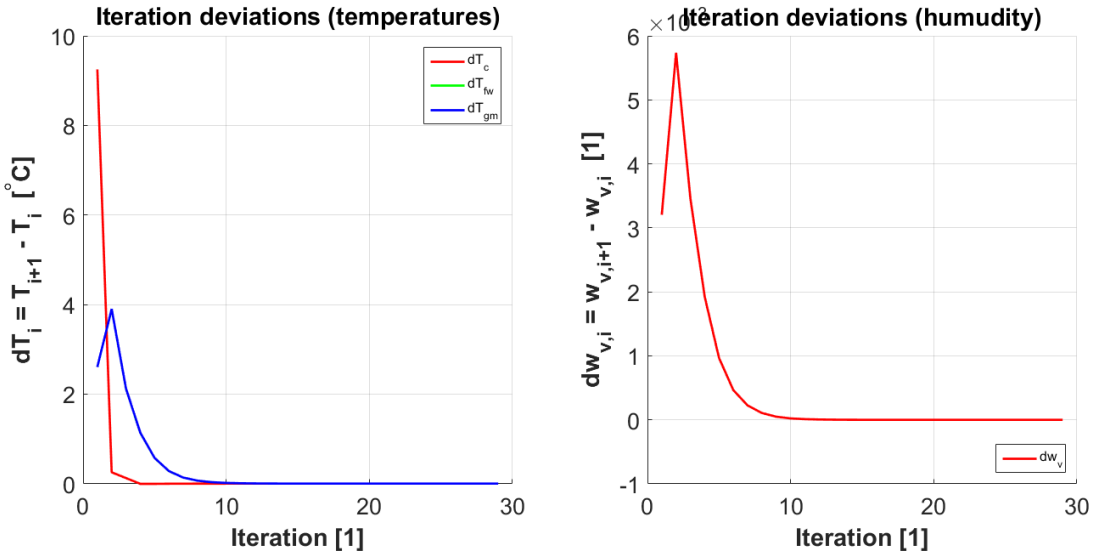
Appendix 9 – Volume condensation rate



Appendix 10 – Air saturation plot

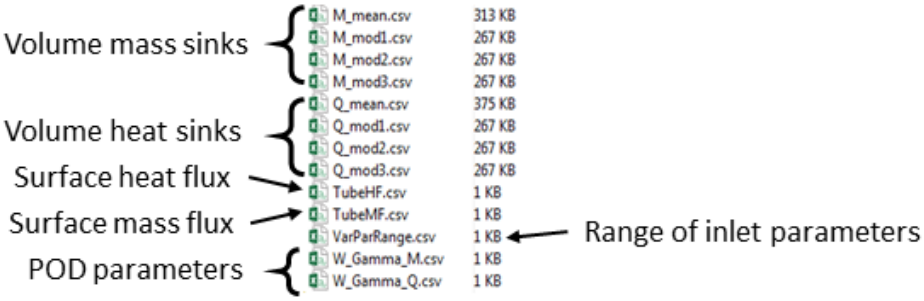


Appendix 11 – Iteration residuals



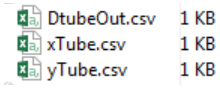
Appendix 12 – Data transfer files

There are 13 files regarding heat and mass sinks.



Files regarding heat and mass sinks

And 3 file regarding tube bundle geometry.



Files regarding tube bundle geometry

Any of those files should not be modified to assure compatibility with STAR-CCM+ loading functions.