

SPECIALIZATION PROJECT 2011

TKP 4550

PROJECT TITLE:

Dynamic model of a crude unit Heat Exchanger Network
at the Statoil Mongstad refinery

By

Alexandre LERUTH

Supervisor for the project: S.Skogestad - J.Jäschke

Date: 16 december 2011

TABLE OF CONTENTS

1. Introduction.....	3
2. Heat Exchanger Network Modeling.....	5
2.1. Shell and tube heat exchangers.....	5
2.2. Heat exchanger model	6
2.2.1. Topology and Assumptions.....	6
2.2.2. Derivation of the state equations.....	7
2.3. Network model	8
2.3.1. Global Heat balance	10
2.4. Dynamic model Simulation.....	11
2.4.1. The S-function.....	11
2.4.2. Heat exchanger model in Simulink	14
3. Data treatment	15
3.1. Volumes and Areas.....	15
3.2. Fluid Heat capacities	18
3.3. temperatures and Flowrates.....	19
3.4. Fluid Densities.....	22
4. Steady-state matching.....	24
5. Model analysis	27
5.1. Heat Exchanger model analysis	27
5.2. Branch model analysis.....	28
5.3. Heat exchanger Network Model Analysis.....	31
6. Conclusion.....	34
7. Bibliography	35
Appendix.....	36
Data Treatment – Temperatures and flows	36
MATLAB © CODE	40

1. INTRODUCTION

This project is about operation of the heat exchanger network for pre-heat of crude oil at the Statoil Mongstad Refinery. The network is such that the crude oil is splitted in several branches where it is heated by heat recovery of several hot distilled products coming from downstream that needs to be cooled. The crude oil is then introduced in a gas fired heater before fractionation.

Actually, energy costs represent a substantial part of the operating costs in a refinery. Optimal operation of this network has been defined as the maximum achievable temperature of the crude oil outlet stream. In other words, it means to maximize the overall heat transfer between hot and cold streams in order to save energy at the gas fired heater unit.

The central objective of our control structure will thus be to reach as near as possible this optimum. The manipulated inputs are assumed to be the split fractions of the crude oil in the several branches. Disturbances are assumed to be the mass flowrates and temperatures of the feed and of hot fluids. The complexity of the network obviously makes the challenge of this project.

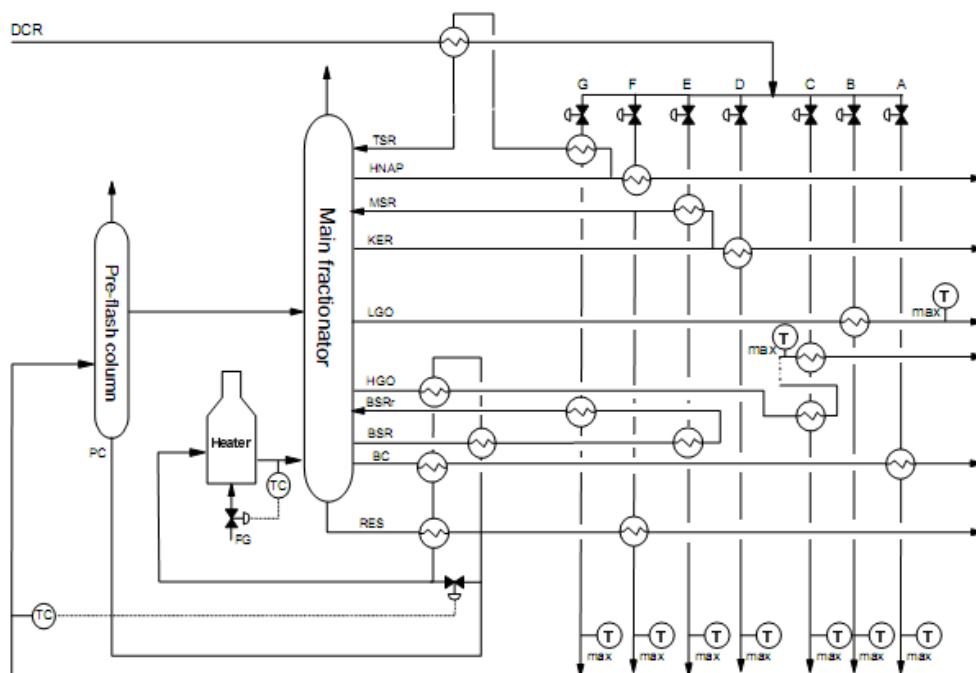


Fig.1. Simplified crude unit overview

Operation of heat exchanger networks is naturally much less studied compared to their design (Glemmestad et al., 1999). Bypass selection for control of heat exchanger networks was investigated by Mathisen et al. (1992). More considerations on utility consumption let him to suggest a method that minimizes it (1994). Boyaci et al. (1996) presented a method based on repeated steady state optimization. Then, Aguilera and Marchetti (1998) introduced a method for on-line optimization and control of heat exchanger networks.

In the Mongstad case, Lid, Strand and Skogestad (2000) introduced on-line optimization of the network. Steady-state mass and energy balance of the whole network (20 heat exchangers) yields their process model. The model is fitted by data reconciliation and optimal split fractions are computed. At that time, this system led to a 2% reduction in energy consumption. However, this method requires many efforts from the operators.

Indeed, using real-time optimization brings difficulties in building and adapting accurate models for complex processes (Chachuat et al., 2009) and the combination of steady state detection, parameter estimation, data reconciliation and solving of a nonlinear optimization problem online (White, 1997) is not very practical for operations.

In this project, a dynamic model of the heat exchanger network will be developed in order to study a new and simplified control method that will be implemented in the following master thesis: self-optimizing control. Self-optimizing control (Skogestad, 2000) is achieved if a constant setpoint policy results in an acceptable loss L without the need to reoptimize when disturbances occur. On the graph below, we see that a loss results when we keep a constant setpoint rather than reoptimizing when a disturbance occurs. A major advantage of this method is that it does not require any model of the heat exchanger network, only simple temperature measurements.

$$L = J(\mathbf{u}, \mathbf{d}) - J_{opt}(\mathbf{d})$$

where

$$\mathbf{u} = \mathbf{f}(\mathbf{c}, \mathbf{d})$$

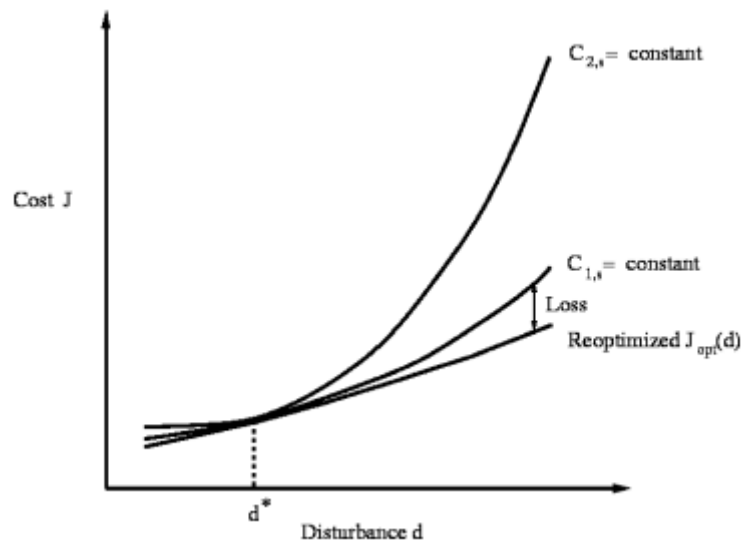


Fig.2. Self-optimizing control

Jäschke and Skogestad (2011) introduced invariants for optimal operation of process systems such as heat exchanger networks. This work follows the master thesis of Daniel Greiner Edvardsen (2011) where a steady-state study using these invariants as self-optimizing variables showed very promising results.

This project will treat only the first part of a larger work which will be continued in the master thesis. This first part was to build a dynamic model of the major branches of the heat exchanger network in Matlab-Simulink, to fit it using plant data received from Mongstad and prepare it for simulations with several control configurations.

2. HEAT EXCHANGER NETWORK MODELING

2.1. SHELL AND TUBE HEAT EXCHANGERS

The Statoil Mongstad preheat train is composed of shell and tube heat exchangers made of steel. This is the more common type of heat exchanger in the petrochemical industry. They handle large flowrates due to their high hydraulic diameter. One set of tubes called the tube bundle contains the first fluid while the second fluid runs over the tubes on the shell side so that heat can be transferred between them. Shell and tube heat exchangers are typically used for high pressure applications due to their shape which insure a strong robustness. We distinguish many types of shell and tubes heat exchanger due to the diversity of internal flow configurations. The most common are made of one, two or four passes on the tube side and only one on the shell side.

Sinnott and Towler (2009) presented some reasons for using shell and tube exchangers:

- The configuration gives a large surface area in a small volume
- Good mechanical layout
- Well-established fabrication techniques
- Can be constructed from a wide range of materials
- Easily cleaned
- Well established design procedures

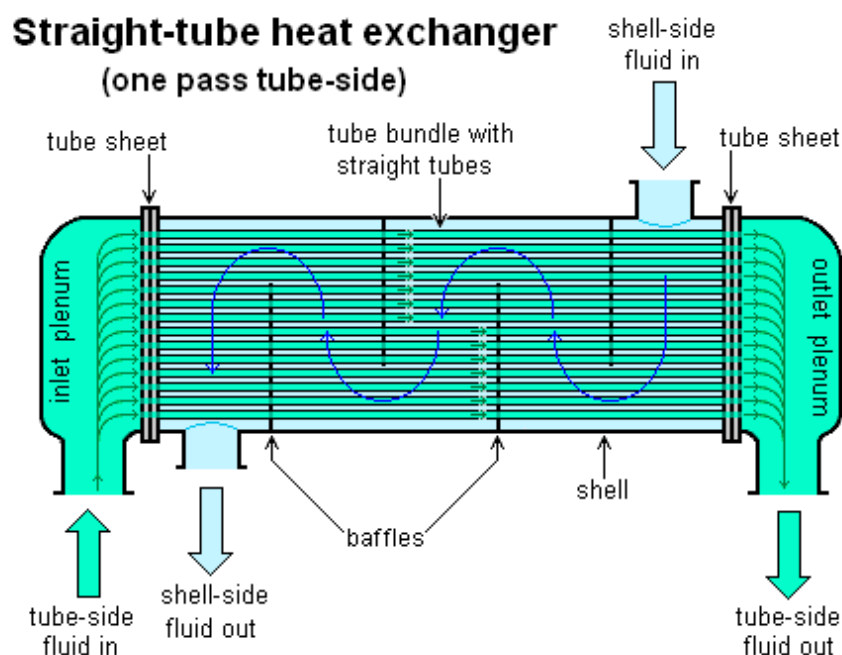


Fig. 3. Shell and tube Heat Exchanger

Baffles are used in shell and tube heat exchangers to direct the fluid on the shell side across the tube bundle. They are perpendicular to the shell and hold the bundle. They also prevent the tubes from sagging and vibrating. Their influence on the flow mixing is to be considered in dynamics studies, especially in pure countercurrent heat exchangers (one pass on each side) which are the units in which the transfer is the most distributed.

With the time, the heat transfer capacity of such units in the crude oil preheat train may be reduced due to fouling. A well-known cause of fouling is asphaltene insolubility which may deposit on the exchange surface areas. On the other hand, we know that the crude oil extracted on the Norwegian Continental Shelf is mainly composed of light fractions.

2.2. HEAT EXCHANGER MODEL

2.2.1. TOPOLOGY AND ASSUMPTIONS

A dynamic model of the heat exchanger network is needed in order to assess controllability. The first step is to build a general dynamic model of shell and tube heat exchanger. Following the work of Mathisen (1994), a flexible lumped multicell model has been developed in order to involve all the important features such as the number of compartments for each fluid (number of elements), fluid heat capacities, heat transfer coefficients (including convective and wall resistances) and wall capacitance.

Since flow configuration has not a major effect on dynamics, a counter-current multi-cell topology has been introduced. As shown on the figure, each internal fluid stream is represented by a serie of N elements of fixed volume. At each step of the counter-current series, hot and cold elements are interconnected by a single and independent wall element. Since the cells are assumed to be ideally mixed, all physical data are assumed to be constant in each cell. The number of cells has been fixed to 10 for all heat exchangers. More complex flow configurations may be easily introduced later as soon as we have more information about the existing heat exchangers. For example, the number of cells could be reduced for heat exchangers with several passes.

Other assumptions are negligible heat loss, negligible pressure drop, constant wall heat capacity, constant fluid densities and equal distribution of areas and volumes over the N cells. The model is thus made of $3N$ states, N hot fluid temperatures, N cold fluid temperatures and N wall temperatures.

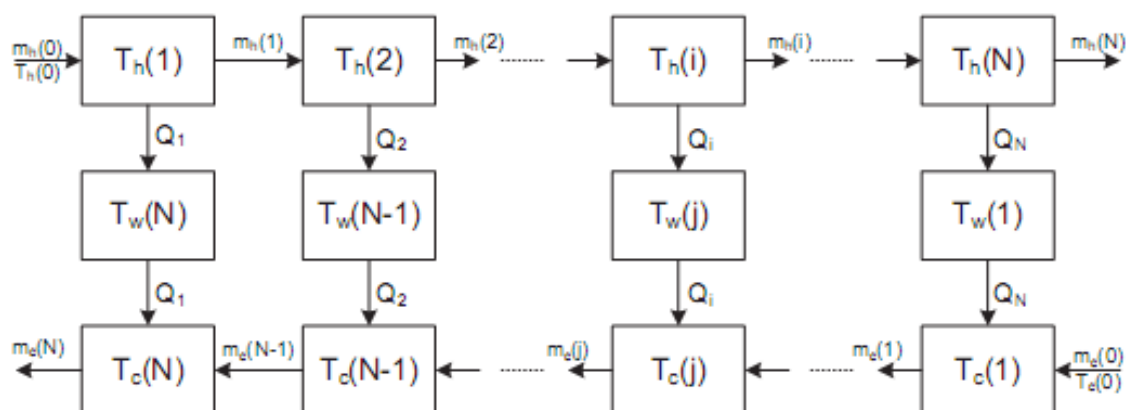


Fig. 4. Heat exchanger model topology

2.2.2. DERIVATION OF THE STATE EQUATIONS

The 2N state equations for the fluid streams are derived from the energy balance on each element :

$$\dot{U}(i) = H(i - 1) - H(i) \pm Q(i)$$

As specific heat capacities is constant in each element :

$$\dot{U}(i) = c_v V(i) \frac{d(\rho(i)T(i))}{dt} = mc_p(T(i - 1) - T(i)) \pm Q(i)$$

As the liquid fluids are assumed to be incompressible and $c_p \approx c_v$:

$$c_p \rho(i) V(i) \frac{dT(i)}{dt} = mc_p(T(i - 1) - T(i)) \pm Q(i)$$

Introducing a simple heat transfer law where heat transfer coefficients represent convective and semi-wall resistances :

$$\begin{aligned} \frac{dT(i)}{dt} &= T(i - 1) - T(i) \pm \frac{hA(i)}{mc_p} \Delta T(i) \frac{m}{\rho V(i)} \\ \Leftrightarrow \frac{dT(i)}{dt} &= T(i - 1) - T(i) \pm \frac{hA}{Nmc_p} \Delta T(i) \frac{mN}{\rho V} \end{aligned}$$

The N state equations for the wall elements are easily obtained by energy balance :

$$\frac{dT(j)}{dt} = (h_n \Delta T_{wh}(j) - h_c \Delta T_{wc}(j)) \frac{A}{\rho_w c_{p,w} V_w}$$

2.3. NETWORK MODEL

The Mongstad crude oil preheat train is shown below. All streams are in complete liquid phase. The crude oil enters the network at the point In1 and is distributed in seven branches A,B,C,D,E,F,G. The heat exchangers are represented in grey while the measurements points (temperature and/or flowrate) are indicated by a circle, white for the crude oil, colored for the hot streams.

The major issue of this project is to prepare a model in order to control as best as possible the distribution of the crude oil in the network, focusing on the several branches. Since the formula used in the self-optimizing method have been developed so far only for parallel branches that split from one single point and gather in another single point, the sub-network indicated by the red box has been studied so far in this project. This sub-network consists of 6 branches from A to F and 11 heat exchangers involved in the crude oil heating.

The rest of the network is a scheme quite similar to serie structure so the direct objective of the control strategy is different than finding an optimal distribution of the crude oil in several branches. Consequently, this second sub-network has less degree of freedom and offers less possibilities of control and influences on the final temperature. However, a study on this second sub-network or, at least, on its influences to the first sub-network will be required to implement successfully the self-optimizing control method.

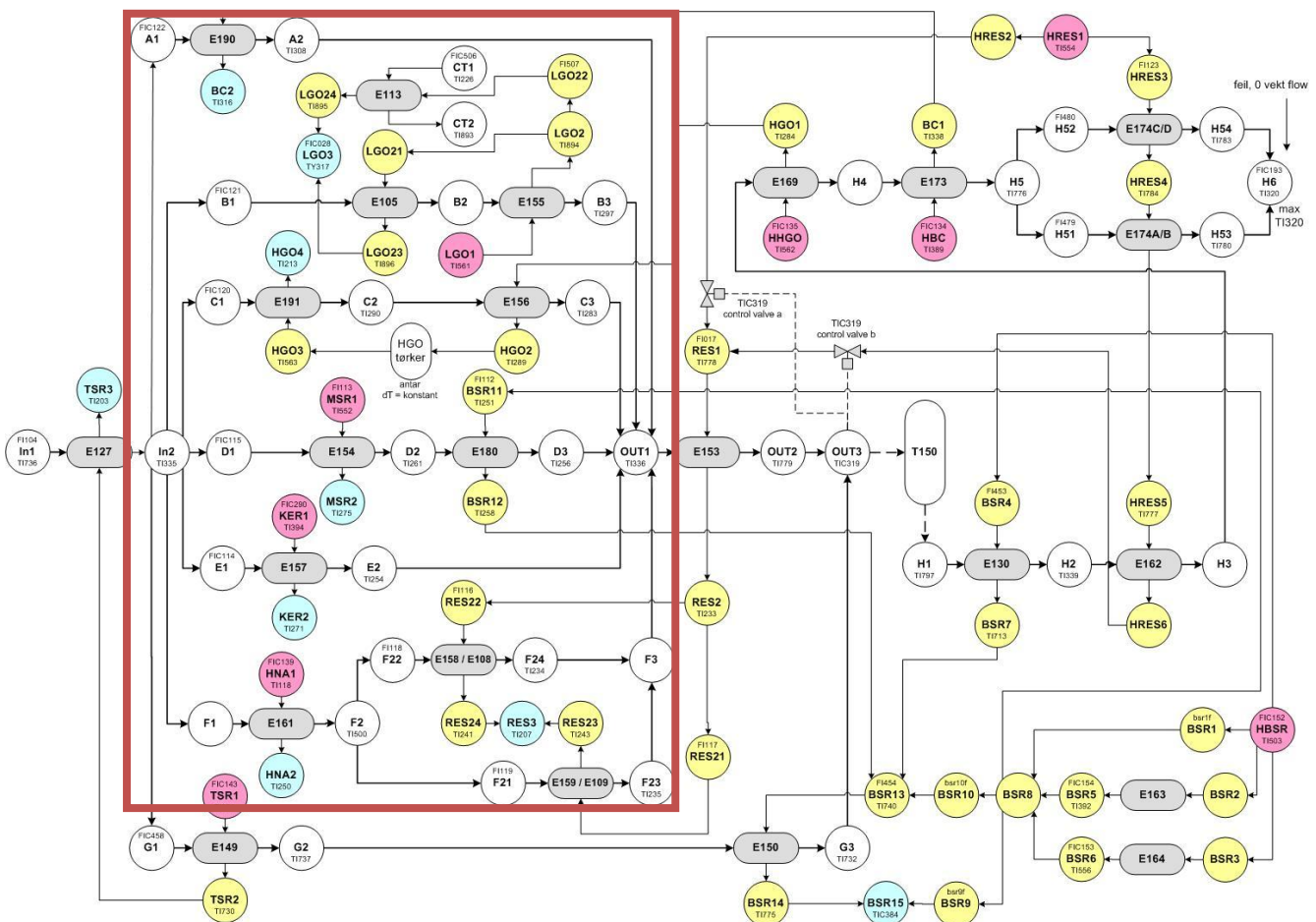


Fig. 5. Statoil Mongstad crude unit Heat Exchanger Network

The heat exchanger network that will be modeled in this project can be described as it is shown on this simplified process flow diagram. Since the same hot fluid **RES** is used in the heat exchangers F2 and F3, we can reasonably simplify even more the network by making a fusion of these two heat exchangers. So, the final network to be modeled in this project will consist of 10 heat exchangers.

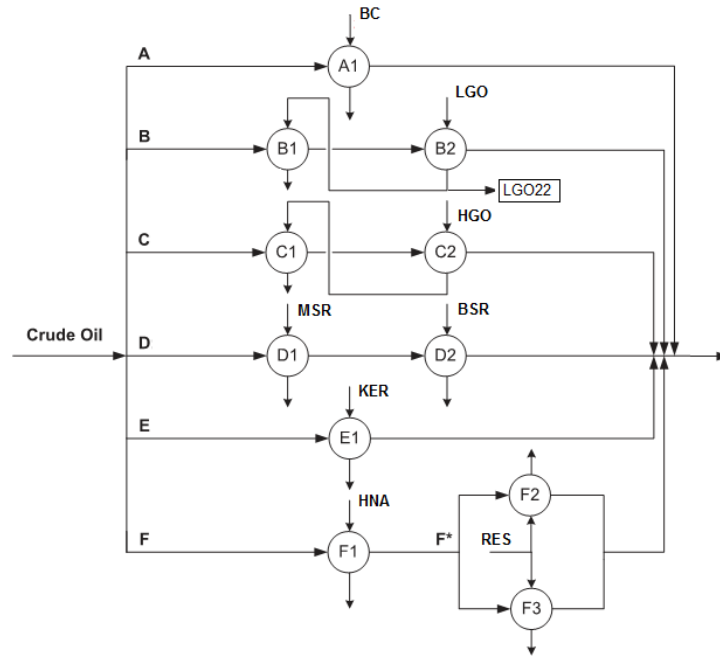


Fig.6. First simplified process flow diagram

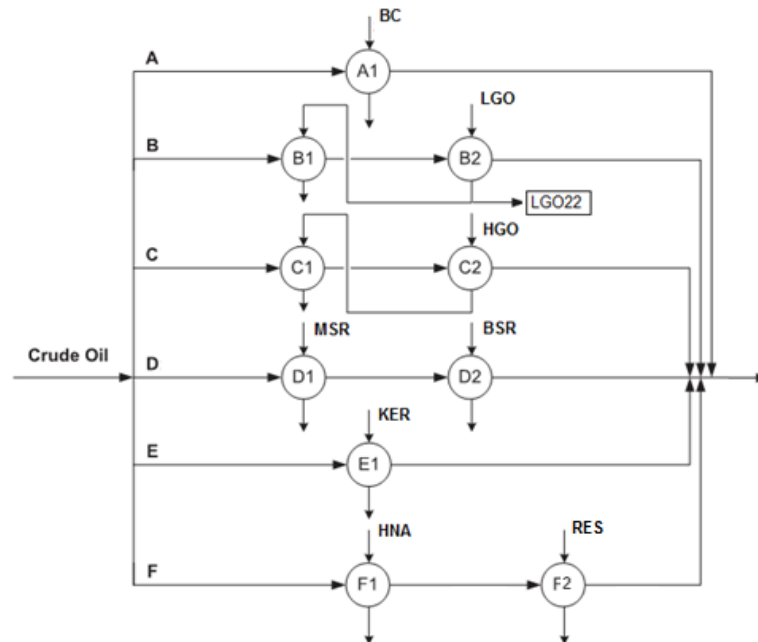


Fig.7. Second simplified process flow diagram

2.3.1. GLOBAL HEAT BALANCE

When the crude oil has been heated by the several branches of the heat exchanger network, the crude oil streams are gathered in a mixer and the outlet total temperature can be computed with the energy balance.

$$\begin{aligned}\Delta H &= \sum_{streams} m_i \Delta h_i \\ \Leftrightarrow \Delta H &= \sum_{streams} m_i \int_{T_{ref}}^{T_i} c_p dT \\ \Leftrightarrow \Delta H &= \sum_{streams} m_i \int_{T_{ref}}^{T_i} (k_{1,crude} T + k_{2,crude}) dT\end{aligned}$$

The formula for the heat capacities is introduced in the section *Data treatment, heat capacities*.

$$\begin{aligned}\Rightarrow \Delta H &= \sum_{streams} m_i \left[k_1 \frac{T^2}{2} + k_2 T \right]_{T_{ref}}^{T_i} \\ \Leftrightarrow \Delta H &= \sum_{streams} m_i T_i \left(k_1 \frac{T_i}{2} + k_2 \right) - m_{tot} T_{ref} \left(k_1 \frac{T_{ref}}{2} + k_2 \right) \quad (1) \\ \text{where} \quad m_{tot} &= \sum_{streams} m_i\end{aligned}$$

On the other hand, we can express this amount of energy as a function of the unknown outlet total temperature T_{tot} .

$$\begin{aligned}\Delta H &= m_{tot} \int_{T_{ref}}^{T_{tot}} c_p dT \\ \Leftrightarrow \Delta H &= m_{tot} \int_{T_{ref}}^{T_{tot}} (k_1 T + k_2) dT \\ \Leftrightarrow \Delta H &= m_{tot} T_{tot} \left(k_1 \frac{T_{tot}}{2} + k_2 \right) - m_{tot} T_{ref} \left(k_1 \frac{T_{ref}}{2} + k_2 \right) \quad (2)\end{aligned}$$

By writing the equation (1)=(2), we can calculate T_{tot} :

$$\sum_{streams} m_i T_i \left(k_1 \frac{T_i}{2} + k_2 \right) = m_{tot} T_{tot} \left(k_1 \frac{T_{tot}}{2} + k_2 \right)$$

This is a second order equation:

$$aT_{tot}^2 + bT_{tot} + c = 0$$

$$\text{where} \quad a = \frac{m_{tot} k_1}{2} \quad ; \quad b = m_{tot} k_2 \quad ; \quad c = -\sum_{streams} m_i T_i \left(k_1 \frac{T_i}{2} + k_2 \right)$$

The physical solution is the temperature which will be positive, we expect the outlet temperature to be around 150-250°C.

$$\begin{aligned}T_{tot} &= \frac{-b + \sqrt{b^2 - 4ac}}{2a} \\ \Leftrightarrow T_{tot} &= \frac{-m_{tot} k_2 + \sqrt{m_{tot}^2 k_2^2 + 2m_{tot} k_1 \sum_{streams} m_i T_i \left(k_1 \frac{T_i}{2} + k_2 \right)}}{m_{tot} k_1} \quad (*)\end{aligned}$$

2.4. DYNAMIC MODEL SIMULATION

The software used in this project is exclusively Matlab – Simulink which offers both flexibility in implementation and strong solving capacities. Simulink is an appropriate environment for this project. It provides an interactive graphical environment and a customizable set of blocks libraries useful for design, simulation, implementation and testing of time-varying systems including control.

2.4.1. THE S-FUNCTION

The shell and tube heat exchanger model has been introduced in Simulink as a S-function (system-function). This mechanism offers to extend the capabilities of the Simulink environment. So an S-function is a computer language description of a Simulink block written in MATLAB or in C, C++, Fortran. S-functions are dynamically linked subroutines that the MATLAB interpreter can automatically load and execute.

Actually, S-functions use a special calling syntax called the S-function API that enables to interact with the Simulink engine. This interaction is very similar to the interaction that takes place between the engine and built-in Simulink blocks. S-functions follow a general form and can accommodate continuous, discrete, and hybrid systems.

An algorithm can thus be implemented in an S-function which will be used as a block added in a Simulink model. After having written the S-function and placed its name in an S-Function block (available in the Functions block library), the user interface can be customized using masking.

The S-function is thus very convenient for creating describing a system as a set of mathematical equations. In this project, S-function were useful because they allow to custom a simple block in the user interface which will be used for all kind of heat exchanger models.

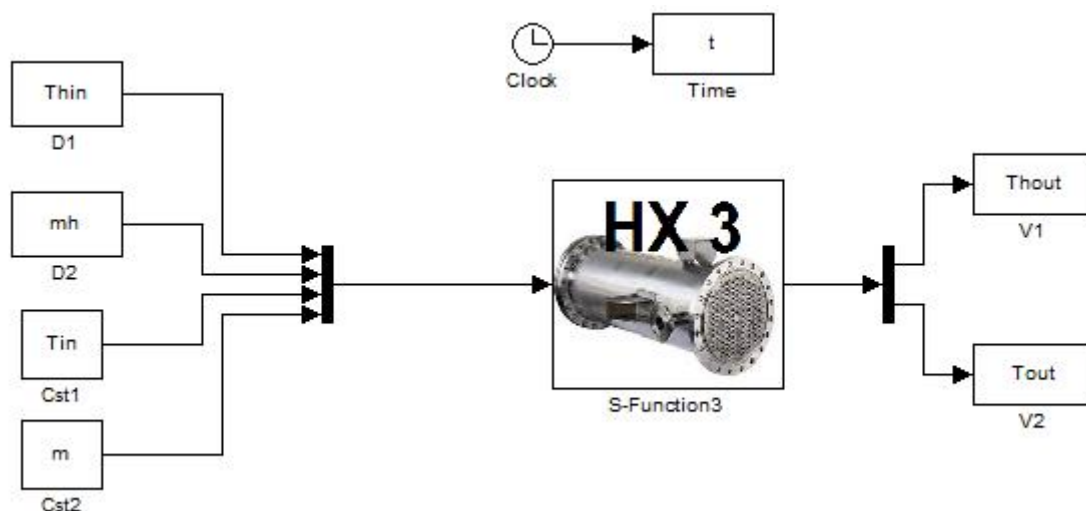


Fig. 8. Heat exchanger model block in the Simulink user interface

It is possible to specify parameter values to be passed to the s-functions using the S-Function block S-function parameters. The order in which the function requires them is to be respected. The parameter values can be constants, names of variables defined in the MATLAB or model workspace, or MATLAB expressions.

It is of interest to understand how S-functions work. This knowledge first requires an understanding of how the Simulink engine simulates a model, including the mathematics of blocks. So, a Simulink block consists of a set of inputs, a set of states, and a set of outputs, where the outputs are a function of the simulation time, the inputs, and the states.

$$\begin{aligned}
 y &= f_0(t, x, u) && \text{(Outputs)} \\
 \dot{x} &= f_d(t, x, u) && \text{(Derivatives)} \\
 x_{d_{k+1}} &= f_u(t, x_c, x_{d_k}, u), && \text{(Update)}
 \end{aligned}$$

where $x = [x_c; x_d]$.

Execution of a Simulink model proceeds in stages. First comes the initialization phase. The engine then enters a simulation loop, where each pass through the loop is referred to as a simulation step. During each simulation step, the engine executes each block in the model in the order determined during initialization. For each block, the engine invokes functions that compute the block states, derivatives, and outputs for the current sample time.

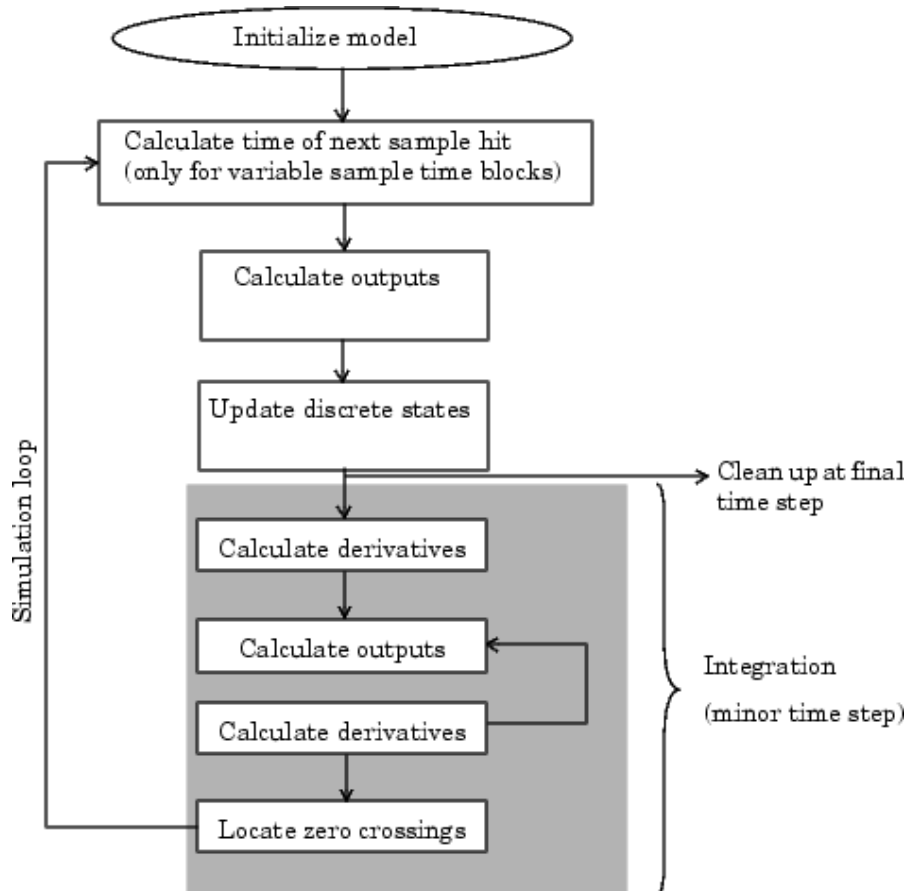


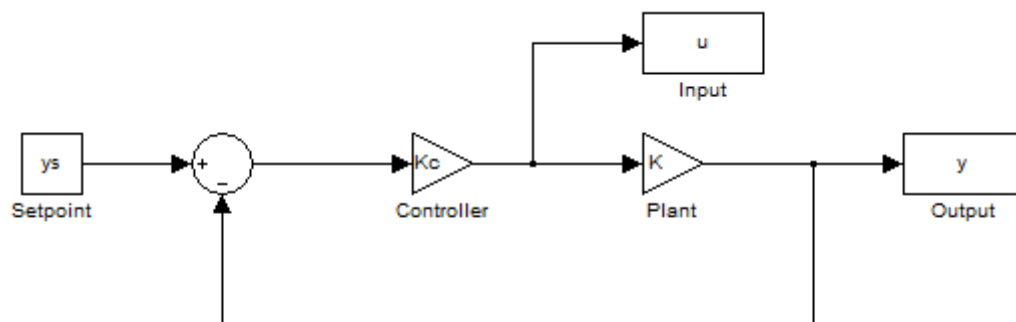
Fig 9. Execution of a Simulink model

The inner integration loop takes place only if the model contains continuous states. The engine executes this loop until the solver reaches the desired accuracy for the state computations. The entire simulation loop then continues until the simulation is complete.

During simulation of a model, at each simulation stage, the Simulink engine calls the appropriate methods for each S-Function block in the model. Tasks performed by S-function callback methods include:

- Initialization :
 - Initializing a simulation structure that contains information about the S-function
 - Setting the number and dimensions of input and output ports
 - Setting the block sample times
 - Allocating storage areas
- Calculation of next sample hit :
 - For a variable sample time block, the next step size is calculated
- Calculation of outputs in the major time step:
 - All the block output ports are valid for the current time step
- Update of discrete states in the major time step:
 - Once-per-time-step activities such as updating discrete states
- Integration:
 - The engine calls the output and derivative or zero-crossing portions of the S-function at minor time steps and so the solvers can compute the state or locate the zero crossings.

A useful concept in s-function is the presence of the flags that directs the engine to the appropriate code in the several steps of the execution. In case of direct feedthrough (the output is controlled directly by the value of an input port signal), a special flag can be used to detect algebraic loops which may force the simulation results of the S-function to not converge. Due to our assumption of incompressibility, it may be a good idea to set the algebraic loop solver diagnostic to warning during the control phase (closing loops).



$$y = K * u \quad ; \quad u = K_c (y_s - y)$$

Fig.10. Simple example of algebraic loop

2.4.2. HEAT EXCHANGER MODEL IN SIMULINK

The heat exchanger network model is then implemented in Simulink.

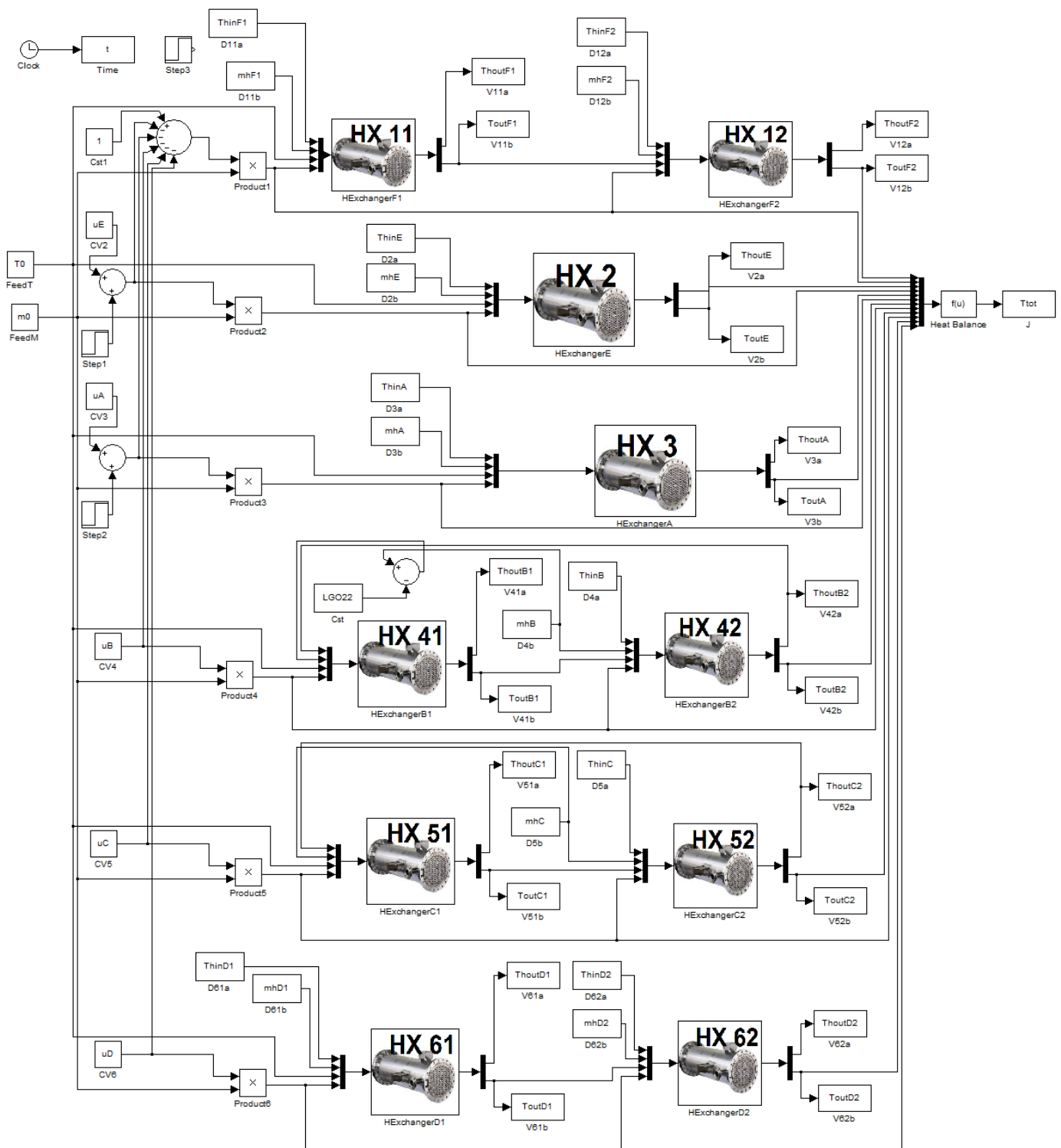


Fig. 11. Heat Exchanger Network Model in Simulink

3. DATA TREATMENT

In this project, many data were provided by Statoil Mongstad and needed to be selected and treated appropriately while some unknown values had to be estimated. This section presents the steps between the data reception and the data introduction in the shell and tube heat exchanger network model.

3.1. VOLUMES AND AREAS

Fortunately, all heat exchanger areas were provided in the data files received from Statoil Mongstad. These values have been directly introduced in the models without any modification.

Unfortunately, some heat exchanger bundle and shell volumes were not provided in the data files received from Mongstad. Simple linear correlations based on the known heat exchangers data have been examined in order to find the best way to estimate the missing data. Hopefully, similarities between the heat exchangers have been observed.

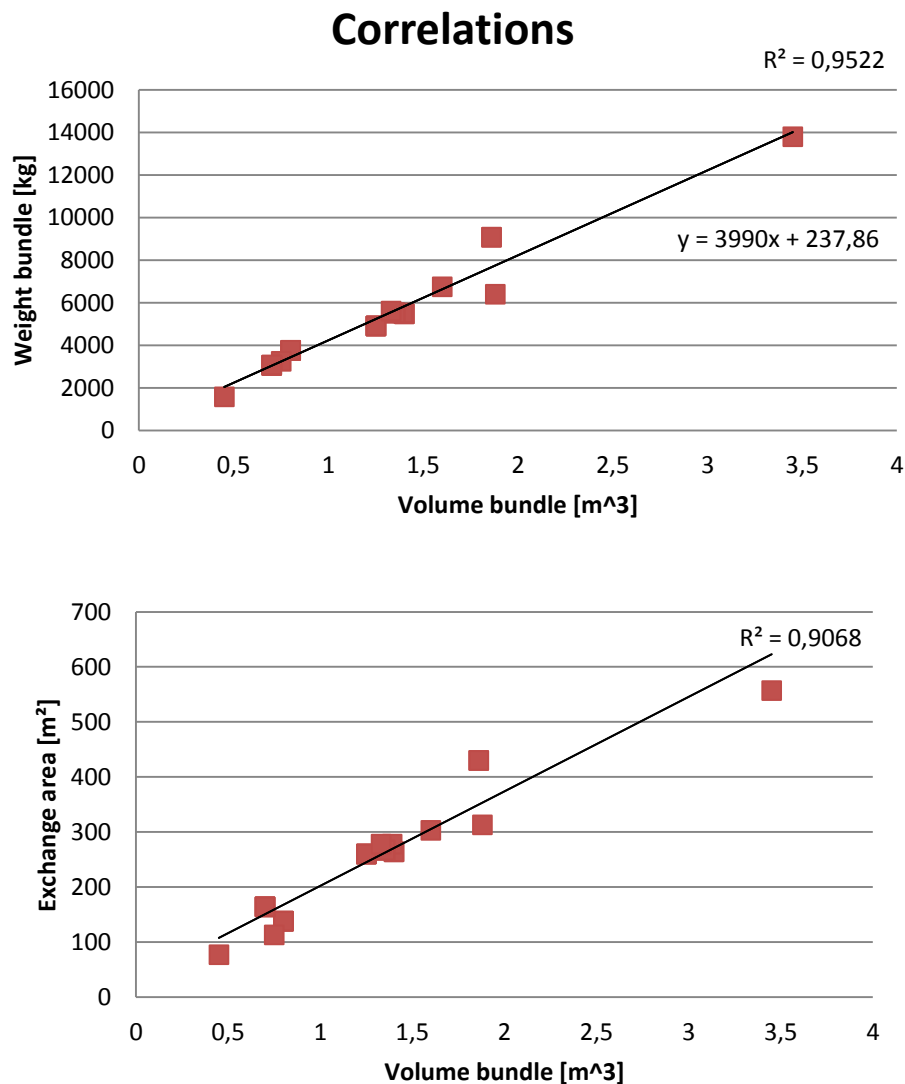


Fig. 12. Correlations for Volume bundle

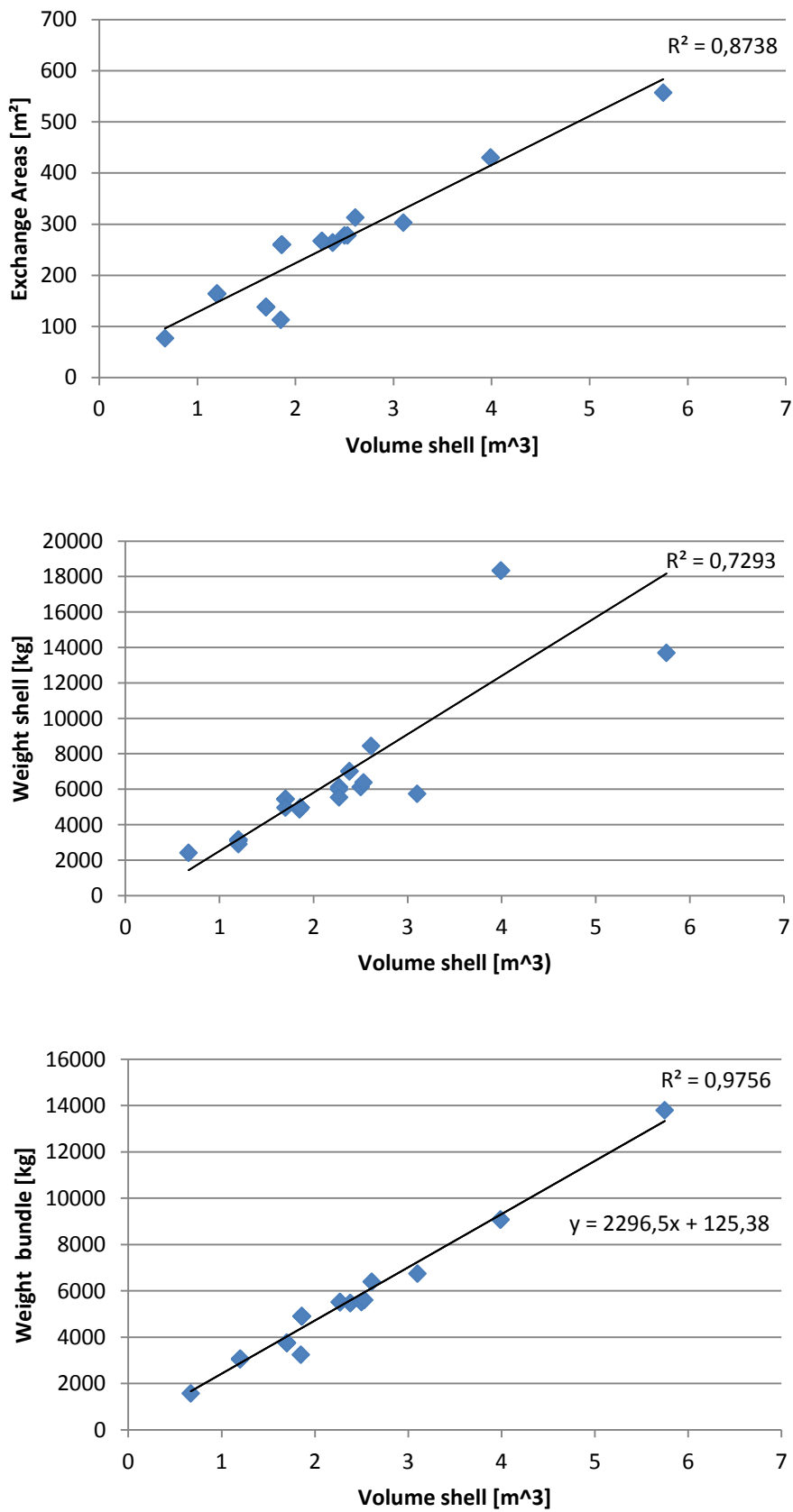


Fig. 13. Correlations for Volume shell

The best correlations (maximum coefficient of determination) were obtained between the weight of the bundle and the volume of the bundle and the shell. So we used them in order to estimate *the missing volumes (in italic in the table below)*.

Heat exchanger	Exchange Area [m ²]	Volume Bundle [m ³]	Volume shell [m ³]	Weight bundle [kg]	Weight shell [kg]
A	138	0,8	1,7	3760	5450
B1	162	<i>0,712</i>	<i>1,287</i>	3080	7430
B2	203	<i>0,960</i>	<i>1,718</i>	4070	3840
C1	264	1,4	2,38	5480	7020
C2	233	<i>1,151</i>	<i>2,049</i>	4830	4890
D1	260	1,25	1,86	4910	4970
D2	313	1,88	2,61	6400	8450
E	164	0,7	1,2	3060	3170
F1	77	0,45	0,67	1580	2420
F2tot (F2+F3)	556 (278+278)	2,72 (1,39+1,33)	5,03 (2,5+2,53)	11130 (5520+5610)	12520 (6130+6390)

The wall weight inside the state equations corresponds to the bundle weight. The material of the wall is steel so the heat capacity of the wall is fixed at 460 J/kgK and the wall density to 7800 kg/m³.

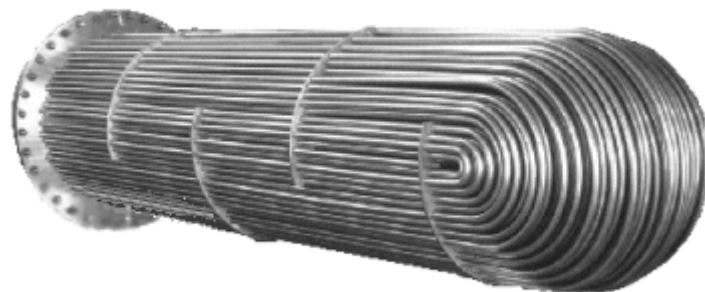


Fig. 14. Heat exchanger bundle

3.2. FLUID HEAT CAPACITIES

The fluid heat capacities at constant pressure were given for each fluid as a linear function of the temperature. So the heat capacity is adjusted in each fluid element using the current temperature (state variable) prior to calculate the temperature derivatives (state equations). We observe that the heat capacity of each fluid varies a lot in its corresponding temperature range so we could not have made the assumption of keeping them constant.

$$c_p = k_1 T + k_2$$

Fluid	k_1 [J/kgK°C]	k_2 [J/kgK]
Crude oil	4,2594	1789,5
RES	3,6378	1779,8
BC	3,9566	1777,1
HGO	3,9802	1792,4
LGO	4,1272	1796,3
KERO	4,4296	1794,1
HNAF	4,9326	1779,1
TSR	5,0218	1779,8
MSR	4,4584	1796,7
BSR	4,0966	1787,7
FCR	3,9018	1784,7

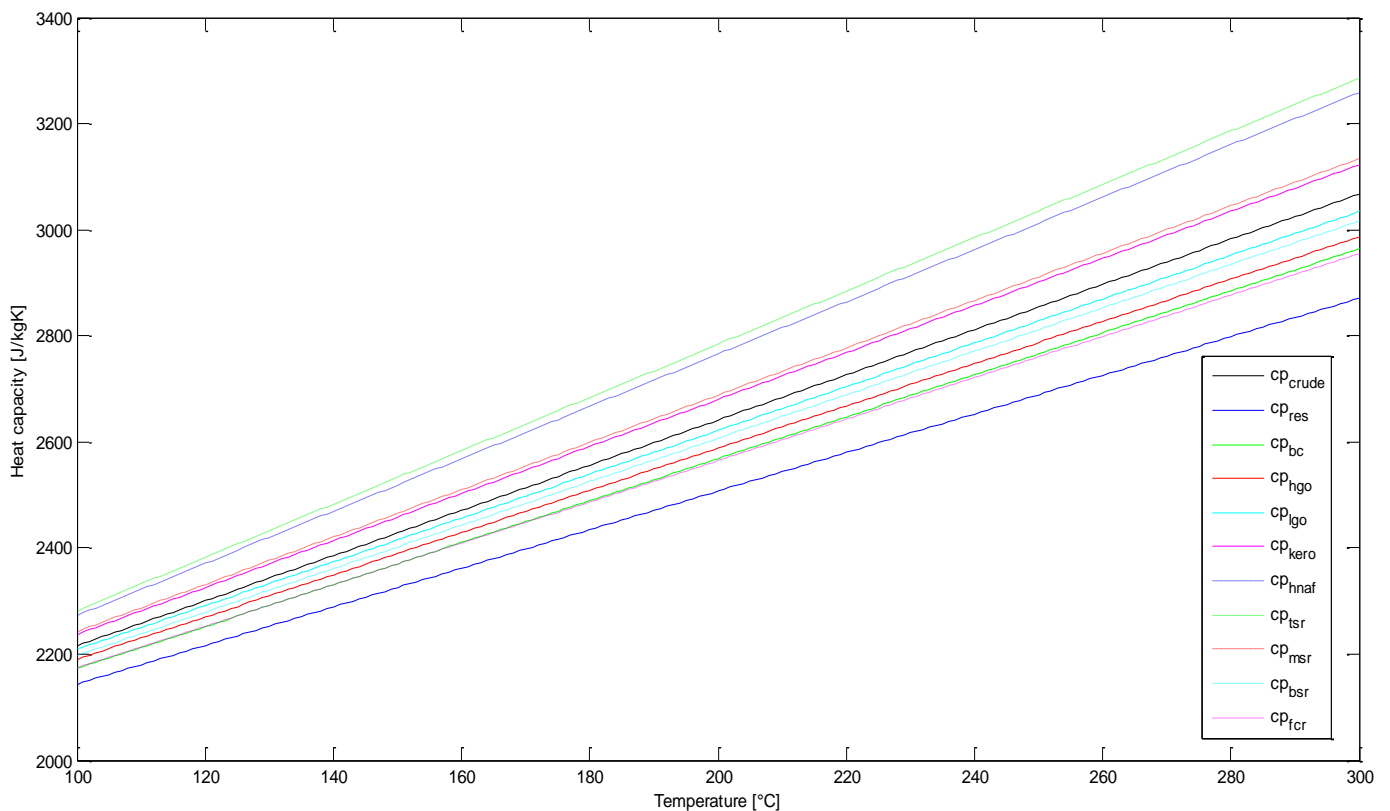


Fig.15. Heat capacities

3.3. TEMPERATURES AND FLOWRATES

The temperatures and flowrates of the network have been given by Statoil Mongstad. Actually, production measurements between the 23 october 2011 at 13:07:25 and 24 october 2011 at 13:06:26 on a 1 minute basis have been provided. Looking at the variations on the feed temperature and other measurements, the values were collected at a point of time where the network seems to be the most stabilized, especially in a thermal perspective (red arrow): the 23 october 2011 at 18:10:36.

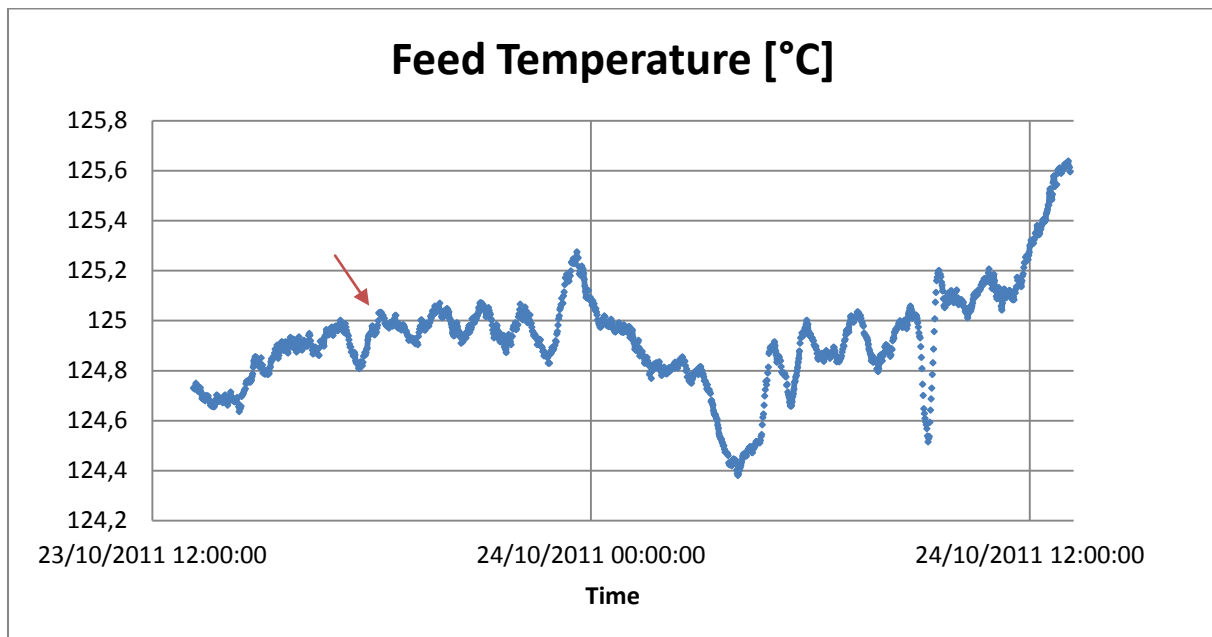


Fig. 16. Feed temperature variations

Mass and energy balances have been used to estimate unknown temperatures and flowrates (some needed streams data have unfortunately not been asked to Mongstad in the list) and simple form of data reconciliation has been used to modify the given measurements as little as possible in order to fulfill the stationary mass and energy balances.

In order to simplify the problem, we assume the uncertainties on the flowrates to be two times higher in percentage than the temperature uncertainties (in percentage on a °C basis). Actually, we observe that the flowrates varies a lot in operations and they may differ a lot compared to a stationary state. We also assume the inlet temperature of the crude oil to be sure and the outlet temperature of the crude oil to be totally uncertain.

Actually, if the desired values for the model differs from the given data, it is probably much more due to the fact that the given data come from ongoing production with all kind of transient effects in the network (different time scales). The measurement errors are probably the main secondary source of deviations.

The data treatment for temperatures and flowrates is detailed branch per branch in the appendix. Nevertheless, we illustrate it here for the more complex branch : branch F. Please refer to the first simplified process flow diagram (Fig. 6) for the notations.

- Exchanger F1 :

$$\begin{aligned}
\Delta H_{F1,crude} &= \Delta H_{HNA} \\
\Leftrightarrow m_F \int_{T_{in}}^{T_{F,inter}} c_{p,crude} dT &= m_{MSR} \int_{T_{HNA,out}}^{T_{HNA,in}} c_{p,HNA} dT \\
\Leftrightarrow m_F \left[T_{F,inter} \left(k_{1,crude} \frac{T_{F,inter}}{2} + k_{2,crude} \right) - T_{in} \left(k_1 \frac{T_{in}}{2} + k_2 \right) \right] \\
&= m_{HNA} \left[T_{HNA,in} \left(k_{1,HNA} \frac{T_{HNA,in}}{2} + k_{2,HNA} \right) - T_{HNA,out} \left(k_{1,HNA} \frac{T_{HNA,out}}{2} + k_{2,HNA} \right) \right]
\end{aligned}$$

- Exchanger F2 :

$$\begin{aligned}
\Delta H_{F2,crude} &= \Delta H_{RES,1} \\
\Leftrightarrow m_{F,2} \int_{T_{F,inter}}^{T_{F2,out}} c_{p,crude} dT &= m_{RES,1} \int_{T_{RES,1,out}}^{T_{RES,in}} c_{p,RES} dT \\
\Leftrightarrow m_{F,2} \left[T_{F2,out} \left(k_{1,crude} \frac{T_{F2,out}}{2} + k_{2,crude} \right) - T_{F,inter} \left(k_{1,crude} \frac{T_{F,inter}}{2} + k_{2,crude} \right) \right] \\
&= m_{RES,1} \left[T_{RES,in} \left(k_{1,RES} \frac{T_{RES,in}}{2} + k_{2,RES} \right) - T_{RES,1,out} \left(k_{1,RES} \frac{T_{RES,1,out}}{2} + k_{2,RES} \right) \right]
\end{aligned}$$

- Exchanger F3 :

$$\begin{aligned}
\Delta H_{F3,crude} &= \Delta H_{RES,2} \\
\Leftrightarrow m_{F,3} \int_{T_{F,inter}}^{T_{F3,out}} c_{p,crude} dT &= m_{RES,2} \int_{T_{RES,2,out}}^{T_{RES,in}} c_{p,RES} dT \\
\Leftrightarrow m_{F,3} \left[T_{F3,out} \left(k_{1,crude} \frac{T_{F3,out}}{2} + k_{2,crude} \right) - T_{F,inter} \left(k_{1,crude} \frac{T_{F,inter}}{2} + k_{2,crude} \right) \right] \\
&= m_{RES,2} \left[T_{RES,in} \left(k_{1,RES} \frac{T_{RES,in}}{2} + k_{2,RES} \right) - T_{RES,2,out} \left(k_{1,RES} \frac{T_{RES,2,out}}{2} + k_{2,RES} \right) \right]
\end{aligned}$$

In these balances, we hopefully have all the data except for m_F which is easily determined by the mass balance:

$$m_F = m_{F,2} + m_{F,3}$$

However, the equations do not match perfectly so reconciliation is needed to obtain stationary values.

m_F [t/h]	T_{in} [°C]	$T_{F,inter}$ [°C]	$T_{HNA,in}$ [°C]	$T_{HNA,out}$ [°C]	m_{HNA} [t/h]
252,54	125,00	136,957	194,664	153,96	72,738

$T_{F2,out}$ [°C]	$T_{RES,in}$ [°C]	$T_{RES,1,out}$ [°C]	$m_{RES,1}$ [t/h]	$T_{F3,out}$ [°C]	$T_{RES,2,out}$ [°C]	$m_{RES,2}$ [t/h]
206,65	244,379	172,484	113,146	199,086	172,207	109,10

For the given values, we observe that:

$$\Delta H_{F1} \ll \Delta H_{HNA}$$

$$\Delta H_{F,2} \approx \Delta H_{RES,1}$$

$$\Delta H_{F,3} > \Delta H_{RES,2}$$

So we introduce a first reconciliation factor $\varepsilon > 0$ such that:

$$T_{r,HNA,in} = T_{HNA,in}(1 - \varepsilon) \quad ; \quad T_{r,HNA,out} = T_{HNA,out}(1 + \varepsilon)$$

$$m_{r,HNA} = m_{HNA}(1 - 2\varepsilon)$$

$$m_{r,F} = m_F(1 + \varepsilon) \quad ; \quad m_{r,F2} = m_{F2}(1 + \varepsilon) \quad ; \quad m_{r,F3} = m_{r,F} - m_{r,F2}$$

$$T_{r,F,inter} = T_{F,inter}(1 + \varepsilon)$$

We optimize the sum of the heat balance absolute errors for ε and obtain:

$$\Delta H_{F1} \approx \Delta H_{HNA}$$

$$\Delta H_{F,2} \approx \Delta H_{RES,1}$$

$$\Delta H_{F,3} > \Delta H_{RES,2}$$

So we introduce a second reconciliation factors $\gamma > 0$ such that:

$$T_{r,RES,2,out} = T_{RES,2,out}(1 - \gamma)$$

$$m_{r,RES,2} = m_{RES,2}(1 + 2\gamma)$$

We optimize again the sum of heat balance absolute errors and obtain:

$$\Delta H_{F1} \approx \Delta H_{HNA}$$

$$\Delta H_{F,2} \approx \Delta H_{RES,1}$$

$$\Delta H_{F,3} \approx \Delta H_{RES,2}$$

So we finally introduce a third reconciliation variable $\delta \geq 0$ such that:

$$m_{r,RES,1} = m_{RES,1}(1 - 2\delta)$$

$$T_{r,RES,1,out} = T_{RES,1,out}(1 + \delta)$$

$$T_{r,F2,out} = T_{F2,out}(1 + \delta) \quad ; \quad T_{r,F3,out} = T_{F3,out}(1 - \delta)$$

And we solve the previous heat balances with the new temperatures and flowrates in order to find, γ and δ :

$$\Rightarrow \varepsilon = 0,00422 \quad ; \quad \gamma = 0,007282 \quad ; \quad \delta = 0,000289$$

The corrected values are :

$m_{r,F}$ [t/h]	T_{in} [°C]	$T_{r,F,inter}$ [°C]	$T_{r,HNA,in}$ [°C]	$T_{r,HNA,out}$ [°C]	$m_{r,HNA}$ [t/h]
253,604	125,00	137,535	193,843	154,610	72,125

$T_{r,F2,out}$ [°C]	$T_{r,RES,in}$ [°C]	$T_{r,RES,1,out}$ [°C]	$m_{r,RES,1}$ [t/h]	$T_{r,F3,out}$ [°C]	$T_{r,RES,2,out}$ [°C]	$m_{r,RES,2}$ [t/h]
205,759	244,379	172,534	113,081	199,028	170,953	110,689

3.4. FLUID DENSITIES

The following table has been provided by Statoil Mongstad

Temp [°C]	Crude	Residue	BPA	HGO	MPA	LGO	KERO	HNA	TPA
15	0,845	0,935	0,874	0,893	0,825	0,859	0,827	0,792	0,788
50	0,821	0,916							
100	0,788	0,89	0,823	0,844	0,768	0,805	0,77	0,73	0,7204
150	0,75	0,864							
200	0,707	0,837	0,759	0,786	0,69	0,737	0,694	0,64	0,631
250	0,657	0,81	0,724	0,754					
300	0,598	0,782	0,684	0,72	0,587	0,66	0,593	0,513	0,499
350		0,751	0,639	0,689					

This leads to the graph below. We observe clearly the results of the crude oil fractionation separating the products according to their volatility (which is related to the density).

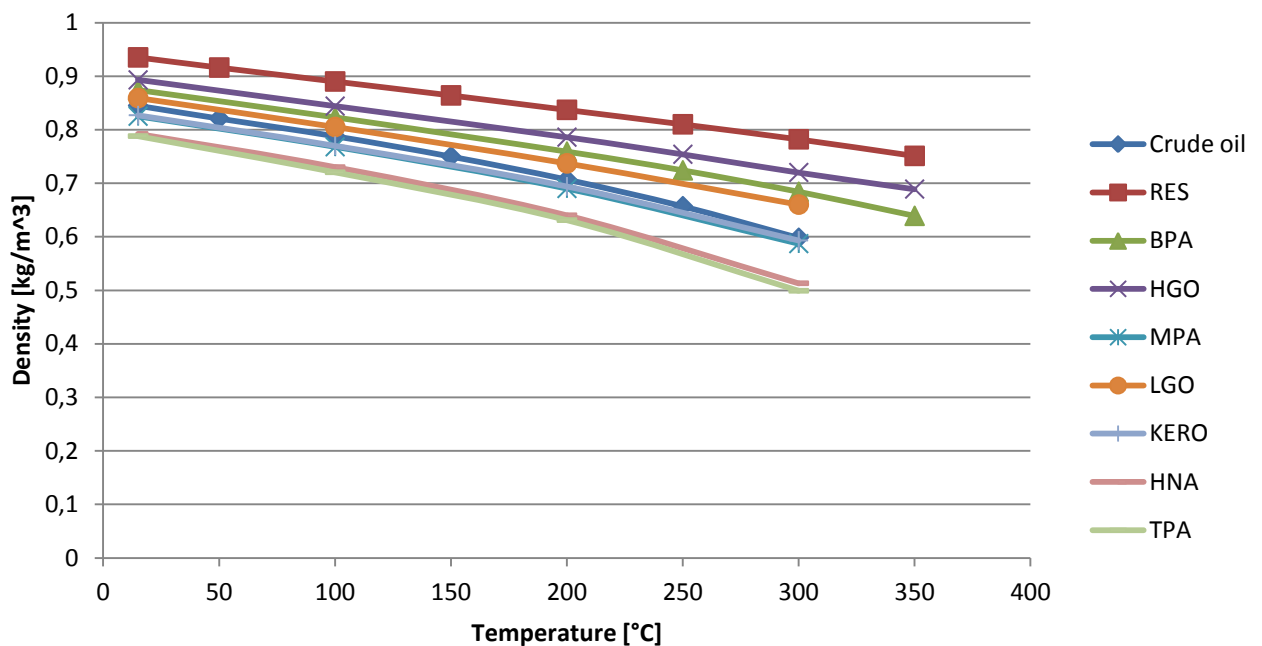


Fig. 17. Fluid densities

Considering the range of temperature for which each fluid is concerned, we find out that the assumption of constant density required by the model is quite strong. Anyway, for each fluid, we fix the density value at the arithmetic mean in the temperature range using linear interpolation. Since we do not have densities for the fluid BC, we assume this fluid to be similar to BPA (=BSR). Actually, BC is similar to BSR in respect the heat capacity values (cfr. Fig. 15) and is also described as a product in the bottom of the fractionation column in Fig.1

The results are shown in the table below.

Fluid	Mean temperature [°C]	Density [kg/m ³]
Crude oil	166,33	0,73596
Residue (=RES)	207,6659	0,83286
BPA (=BSR)	256,221	0,719023
HGO	195,0137	0,788892
MPA (=MSR)	199,6211	0,690296
LGO	223,4688	0,718929
KERO	211,1618	0,682727
HNA	174,2267	0,663196
BC	231,5176	0,736938

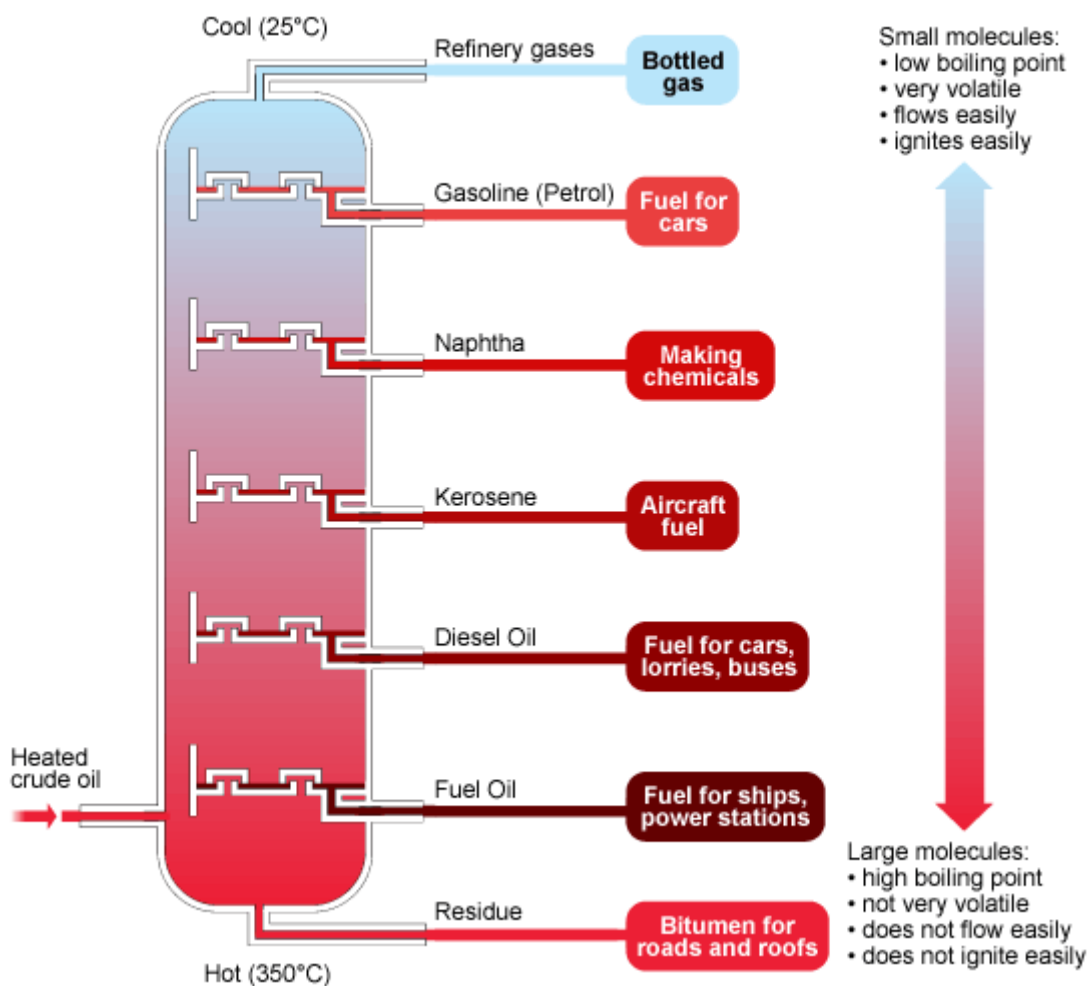


Fig. 18. Fractionation illustration

4. STEADY-STATE MATCHING

Now that we have estimated all stationary values of the heat exchanger network, we can adjust each heat exchanger of the model. This procedure corresponds to the identification of the heat transfer coefficients which are the last unknown model variables. Since the heat transferred in each heat exchanger is a single variable to be adjusted, we decide to use a single variable coefficient by unit. We thus assume the heat transfer coefficient to be the same value on both streams (hot and cold).

As said previously, this simplified heat coefficient includes convective and wall resistances. That's why its value should normally be a bit higher than what can be found in the literature for physical heat coefficients (using heat transfer correlations). The global heat exchanger heat transfer coefficient U can be simply estimated like this:

$$\frac{1}{U} = \frac{2}{h} \Leftrightarrow U = \frac{h}{2}$$

We can expect U values in the range 50-500 W/m^2K so we expect h values in the range 100-1000 W/m^2K .

The heat exchanger state variables are initialized with linear temperature profiles such that:

$$T_{hot}(i) = T_{hot,in} + \frac{i-1}{N-1} * (T_{hot,out} - T_{hot,in})$$

$$T_{cold}(j) = T_{cold,in} + \frac{j-1}{N-1} * (T_{cold,out} - T_{cold,in})$$

$$T_{wall}(j) = \frac{T_{hot}(N+1-j) + T_{cold}(j)}{2}$$

Each branch is simulated separately and the heat transfer coefficients are easily adjusted manually to obtain the stationary heat transfer calculated previously.

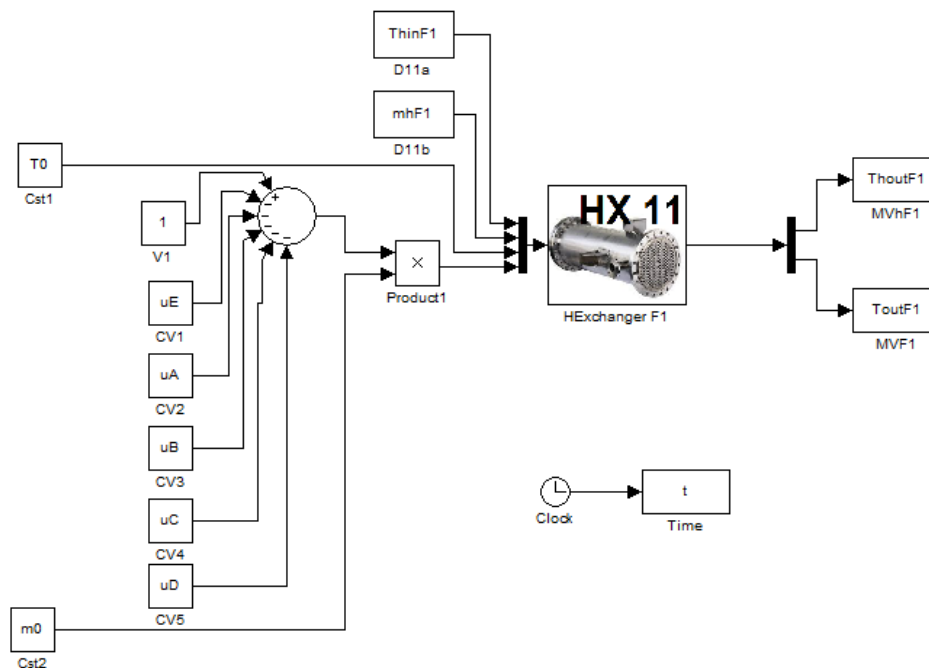


Fig. 19. Model for steady-state matching

In this graph, the outlet temperatures of the heat exchanger F1 are plotted for several heat transfer coefficient.

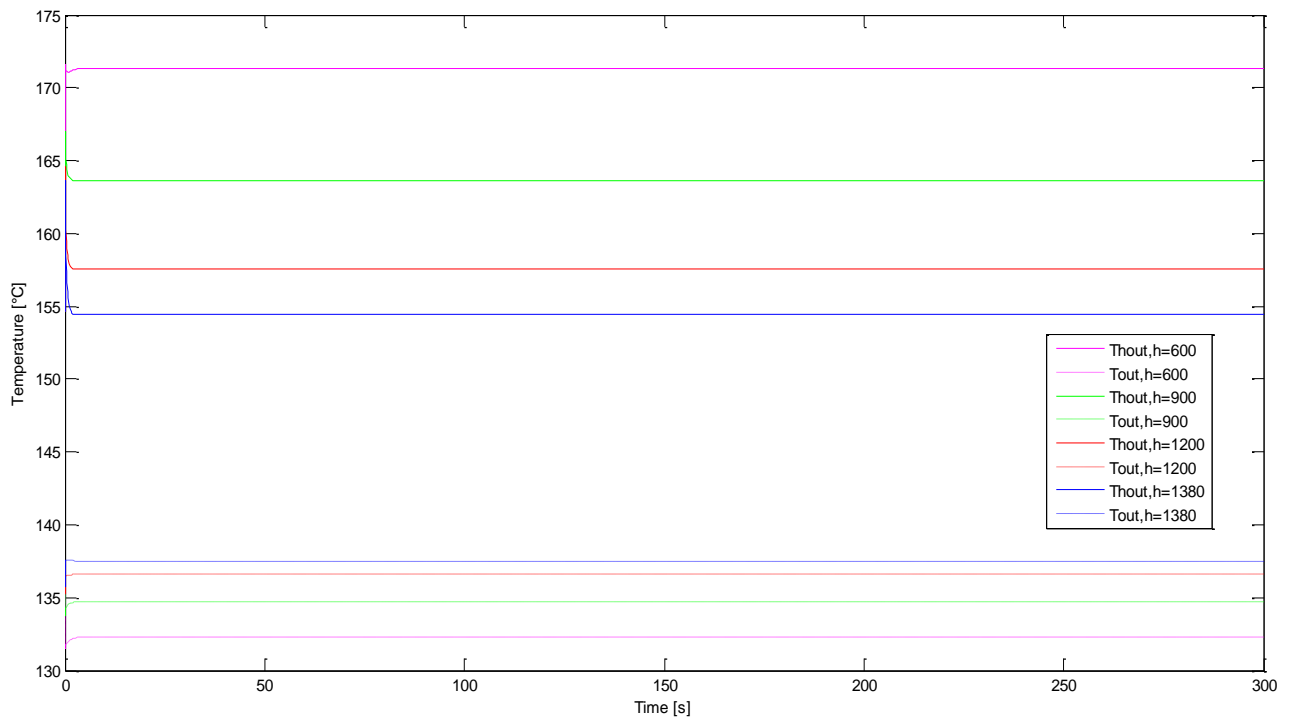


Fig. 20. Steady-state matching

The heat transfer coefficients values from this simulation procedure are listed in table below.

Heat exchanger	Heat transfer coefficient h
A	1902
B1	1189
B2	713
C1	1565
C2	1565
D1	1250
D2	383
E	1976
F1	1381
F2	1348

The results are very much higher than expected. The reason may be due to the strong assumptions on density, the given heat exchanger data or, especially the simplified form of the flow configuration (and the volume distribution).

Once the heat transfer coefficients are tuned, we can re-write the initialization file for the temperature profiles by extracting them out of each heat exchanger with a sufficient simulation time (steady-state).

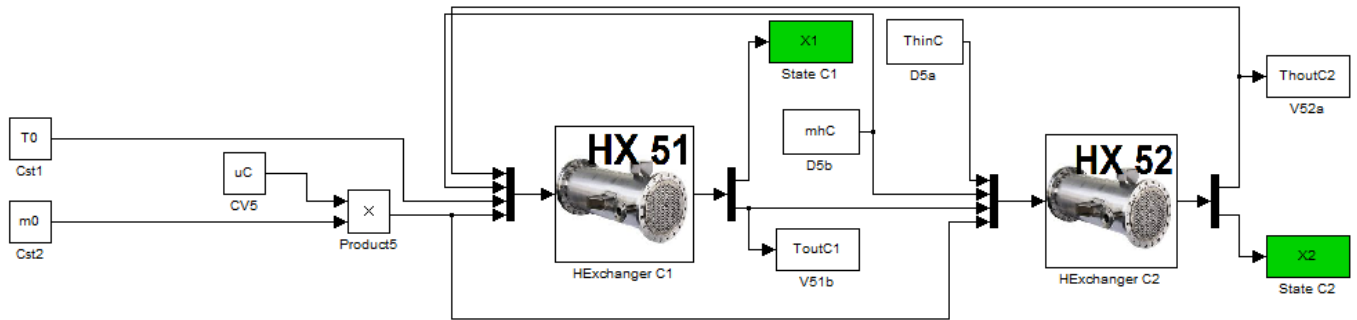


Fig. 21. Model for state variables extraction

The S-functions are modified like this (compared to their code in the appendix):

- HEXchanger C1 (HX 51) :

```
flag==3 → sys(:,1) = x;
flag==0 → sys = [3*N, 0, 3*N+1, 4, 0, 0];
```

- HEXchanger C2 (HX 52) :

```
flag==3 → sys(2:3*N+1,1) = x;
flag==0 → sys = [3*N, 0, 3*N+1, 4, 0, 0];
```

The temperatures profiles obtained are very close to the linear profiles assumed before.

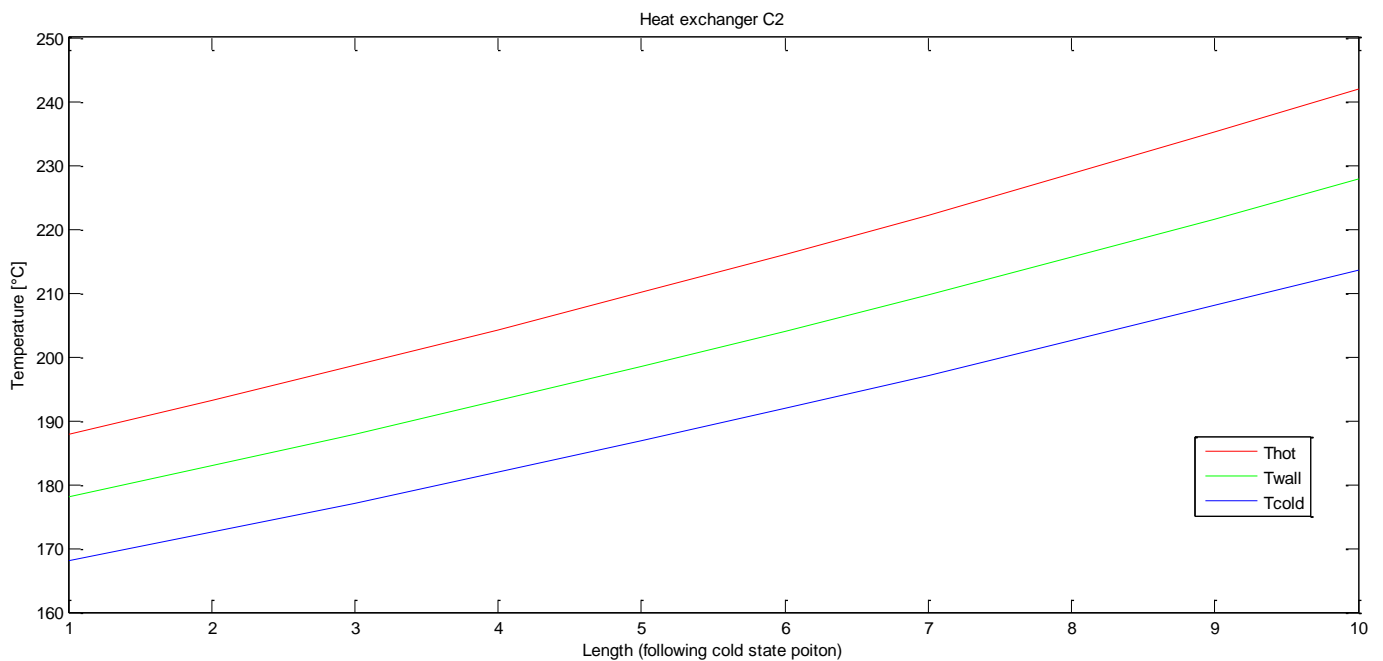


Fig. 22. Internal temperature profile at steady-state

5. MODEL ANALYSIS

It is of great importance to verify that the heat exchanger network dynamic model is physically coherent. We have to analyse how it behaves in respect to what we would expect from the real plant, especially in the time-scale dynamics for which we want to design a new control configuration.

5.1. HEAT EXCHANGER MODEL ANALYSIS

Exchanger F1 is taken as an example.

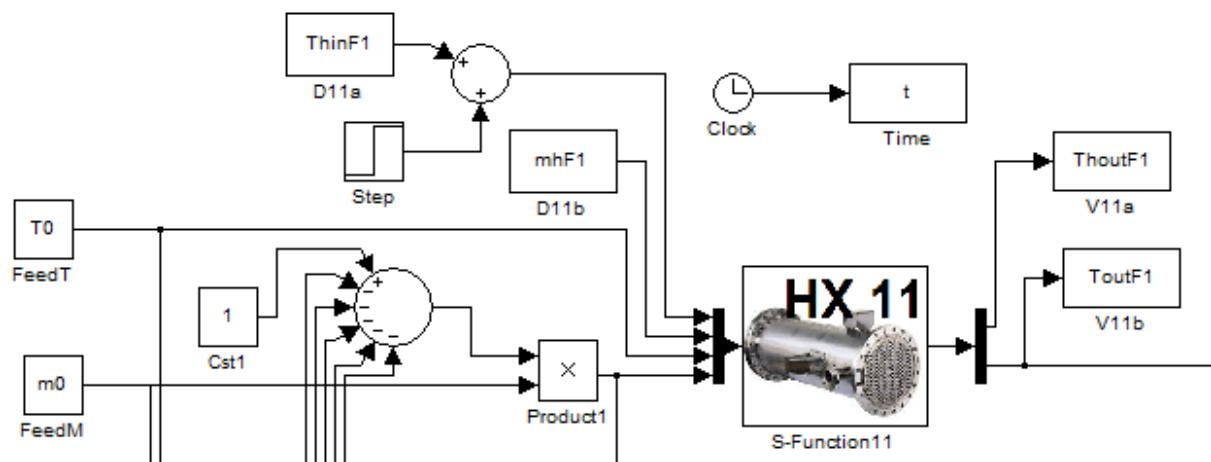


Fig. 23. Heat exchanger model analysis in Simulink

Step change in a inlet temperature:

At time = 20s, ThinF1 passes from 193.843°C to 203.843°C. The hot outlet temperature grows as expected. The heat transfer grows as well so the cold outlet temperature grows. The dynamics is very fast on both sides. Actually, the volumes of the exchanger are very small compared to the flows.

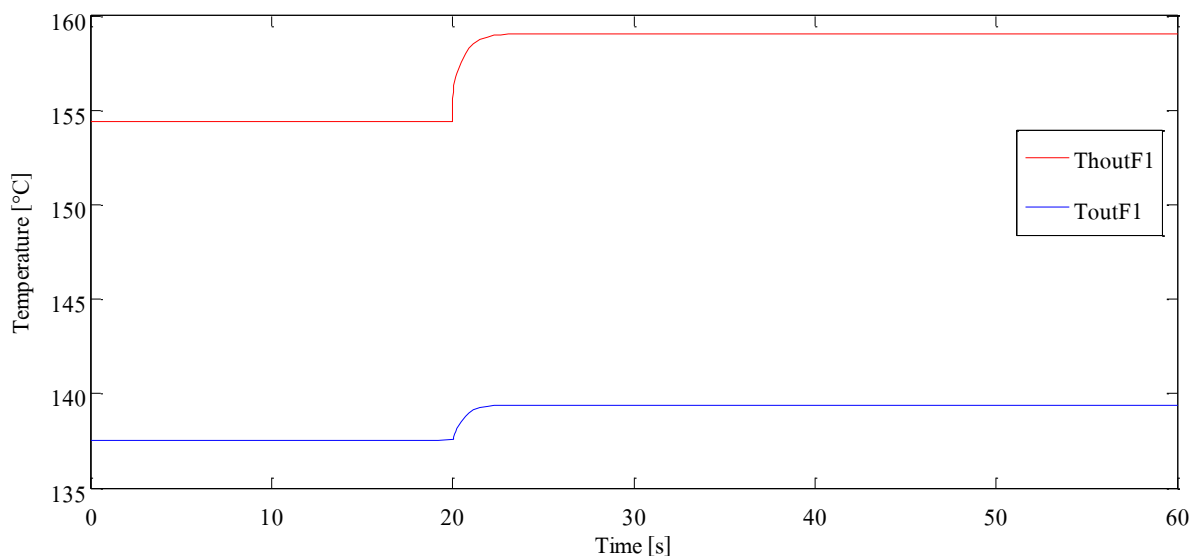


Fig. 24. Step change in ThinF1

Step change in a inlet flowrate:

At time = 20s, m_{inF1} passes from 20.0347 kg/s to 23.0347 kg/s. The same effects happen. We notice the direct change on the hot outlet temperature induced by the assumption of incompressibility. As expected, the heat transfer does not grow so much compared to the previous case.

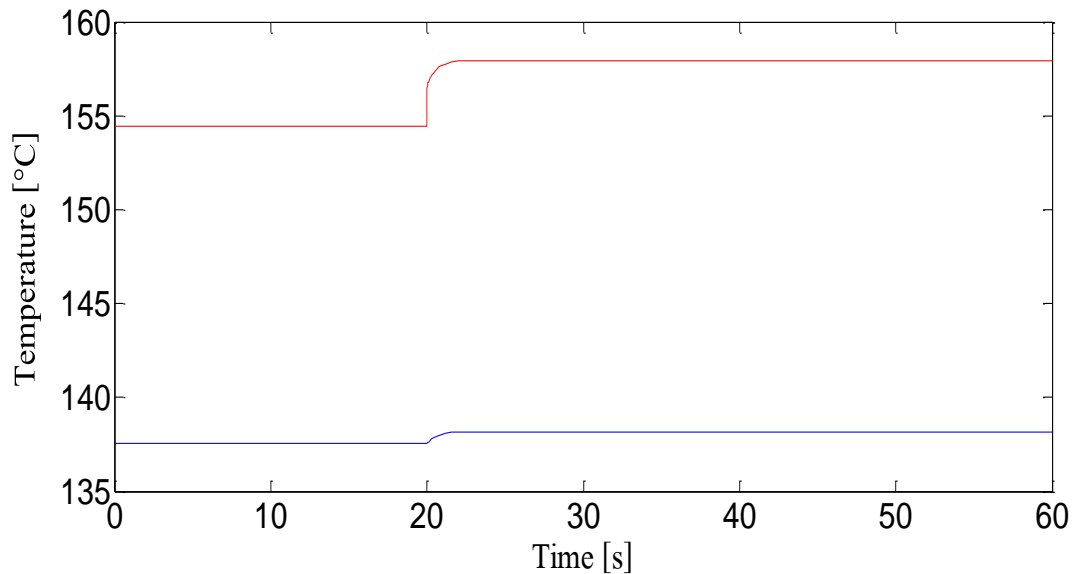


Fig. 25. Step change in m_{inF1}

5.2. BRANCH MODEL ANALYSIS

The Branch C is taken as example.

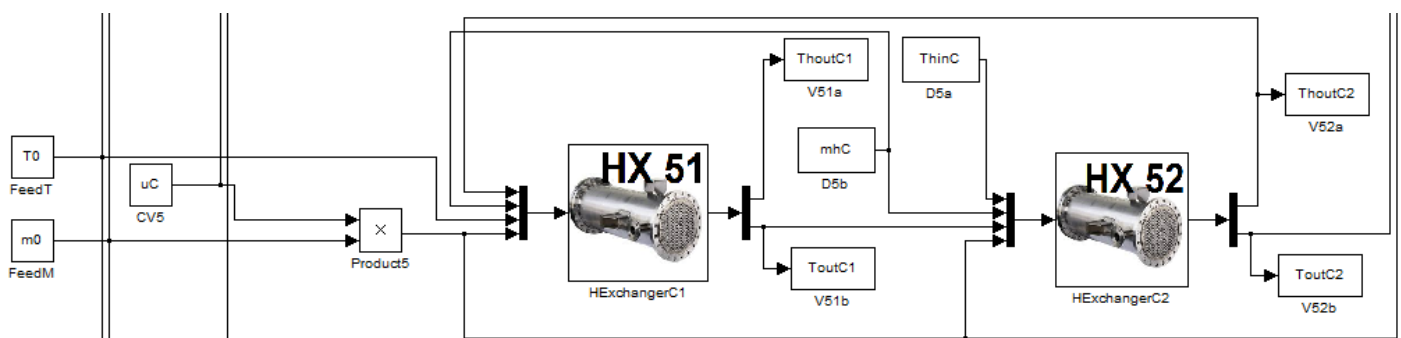


Fig. 26. Branch model analysis in Simulink

Step change in the hot stream inlet temperature:

At time=20s, ThinC passes from 248.7734°C to 228.7734°C. The two exchangers have together such a capacity that the last temperature of the hot streams (outlet of C1) has changed of 3°C only. The heat transfer has been reduced, in both heat exchangers with almost the same intensity.

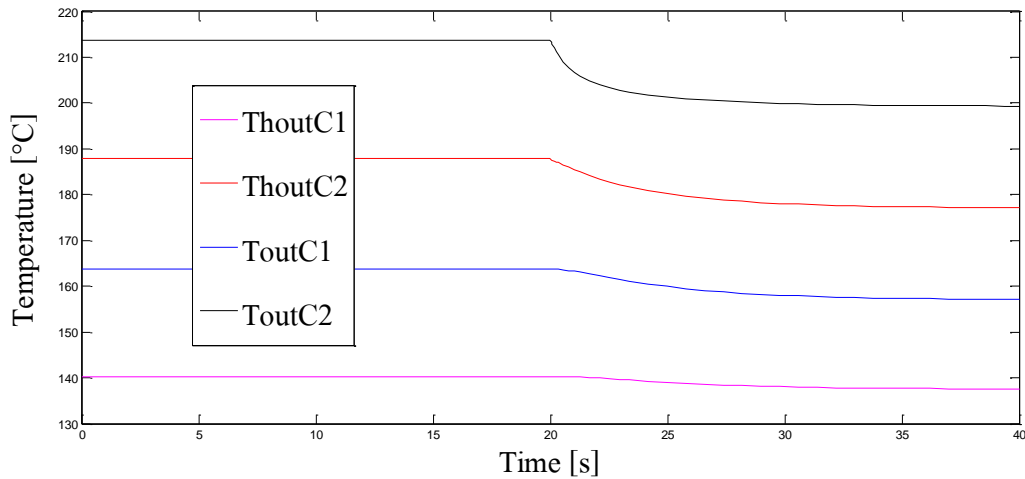


Fig. 27. Step change in ThinC

Step change in the hot stream inlet flowrate:

At time=20s, mhC passes from 26.9653 kg/s to 20.9653 kg/s. We observe that the temperature of the hot fluids reduces very much in the second heat exchanger (the first to be crossed by the hot fluid) so the heat transfer on this unit tends to be conserved. The heat transfer reduces mainly on the first exchanger at lower temperatures where the temperature difference is lower. Considering what makes the driving force in heat transfer, this behavior is physically consistent. We observe that the temperature difference between hot and cold fluid diminishes on the first heat exchanger while it grows a bit on the second one.

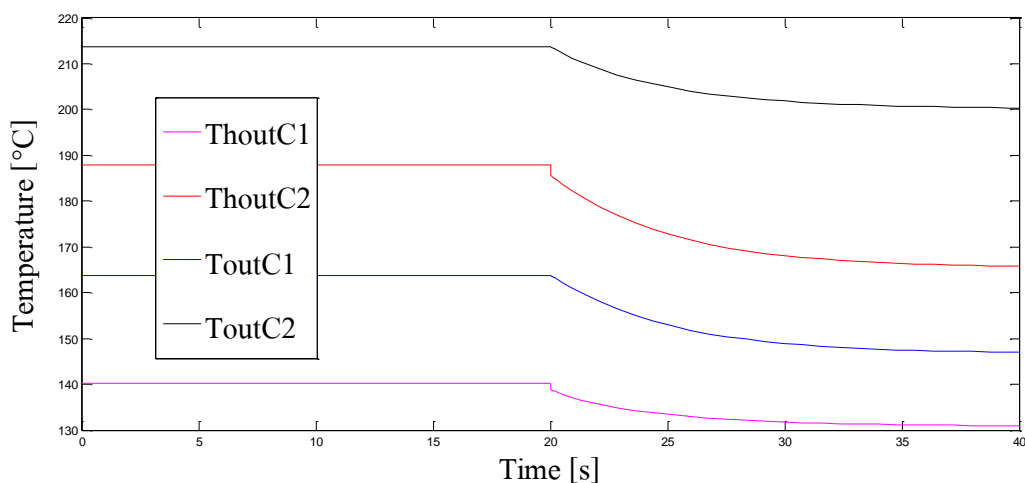


Fig. 28. Step change in mhinC

Step change in the feed flowrate:

At time=20s, the feed of crude oil grows of 10%. As expected, all temperatures outlet reduces. Both exchangers reduce the heat transfer. The hot fluid outlet temperature in the first exchanger is really close to the temperature of the feed of crude oil so the margin to maintain heat transfer on this heat exchanger is really weak compared to the second heat exchanger at the hot temperature. This is physically consistent.

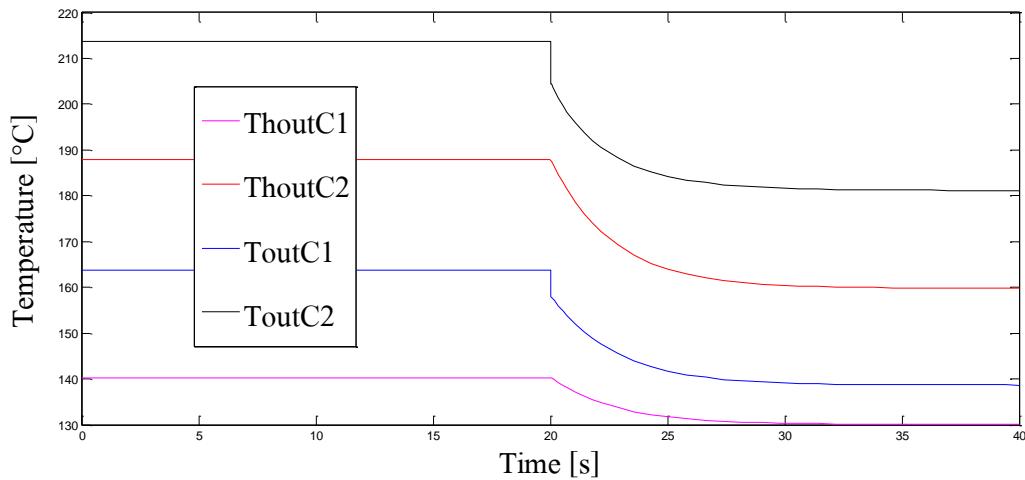


Fig. 29. Step change in the feed mass flowrate – Branch C

Step change in the feed temperature:

At time=20s, the temperature of the feed of crude oil passes from 125°C to 140°C. We observe major influences on the first heat exchanger in term of dynamics (fast) and intensity (temperature grows). This is, again and hopefully, physically consistent.

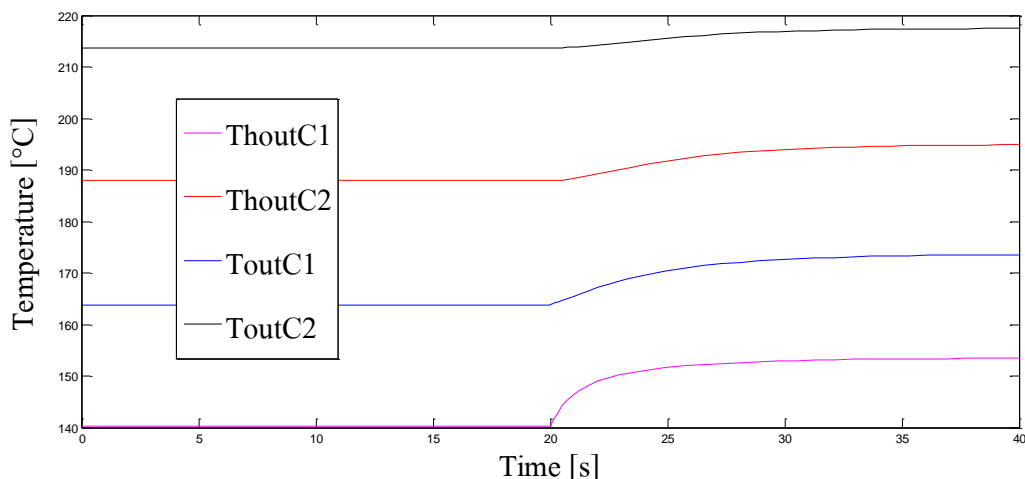


Fig. 30. Step change in the feed temperature – Branch C

5.3. HEAT EXCHANGER NETWORK MODEL ANALYSIS

The complete model is analysed in Simulink (cfr. Fig. 11).

Step change in the feed temperature:

At time=20s, the feed temperature passes from 125°C to 135°C. The fastest branch is E, then comes B and D, A and finally F and C where the heat exchangers are the largest. This information will play a major role in the control implementation, especially in case of cascade control or cascade tuning.

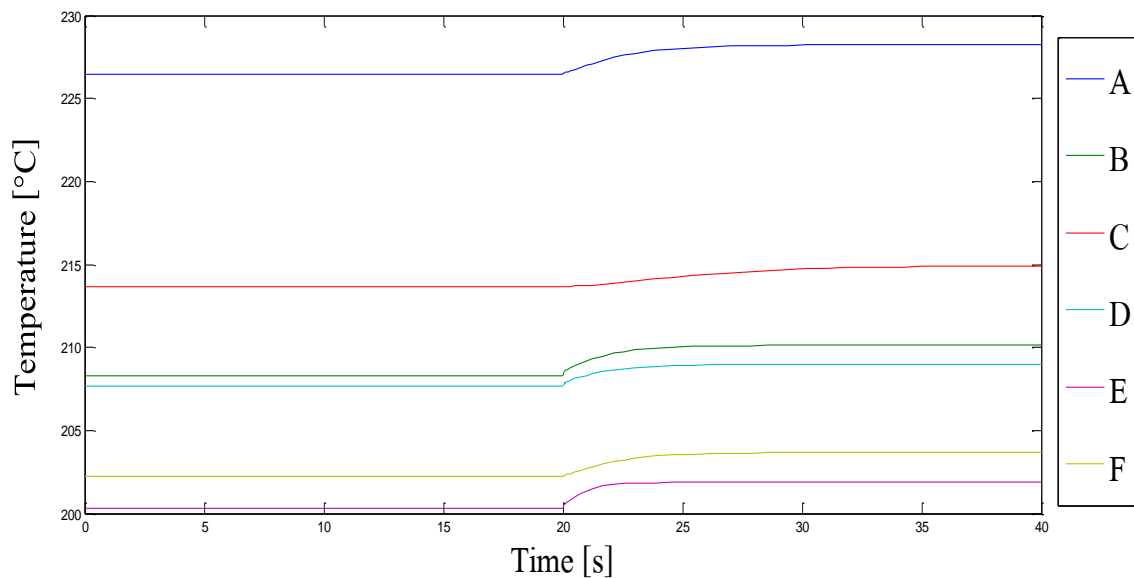


Fig. 31. Step change in the feed temperature – Branches

The effect on the total outlet temperature is plotted below. The temperature grows is of 3°C only so the heat transfer has been reduced as expected (less driving force in the heat exchangers). The global dynamics seems to be quite fast.

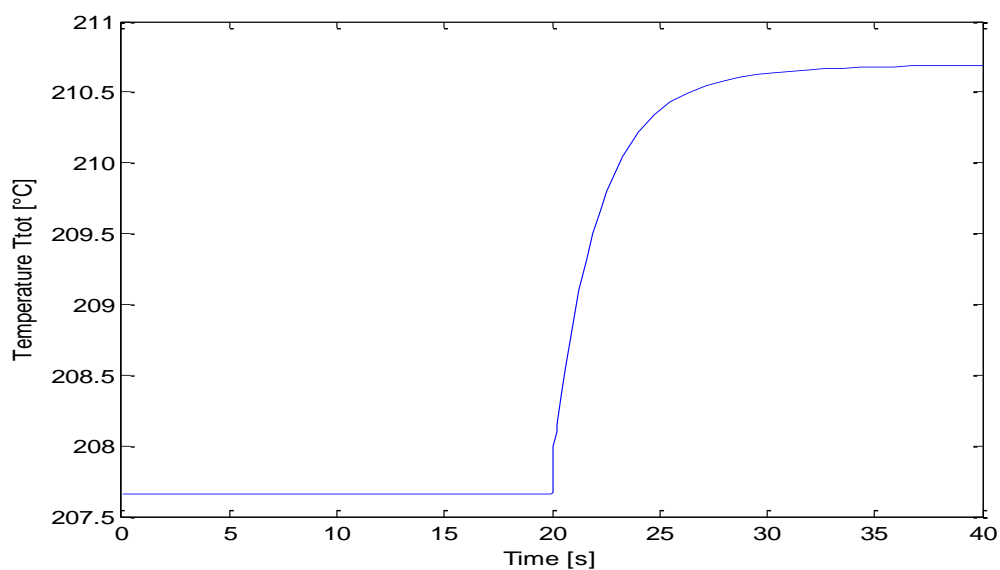


Fig. 32. Step change in the feed temperature - Ttot

Step change in the feed flowrate:

At time=20s, the feed flowrate reduces from 254.2839 kg/s to 234.2839 kg/s. We observe the same sequence in a fastness perspective. The impact of the step is strong due to the direct response (which is due to the incompressibility assumption). This may generate algebraic loop in control.

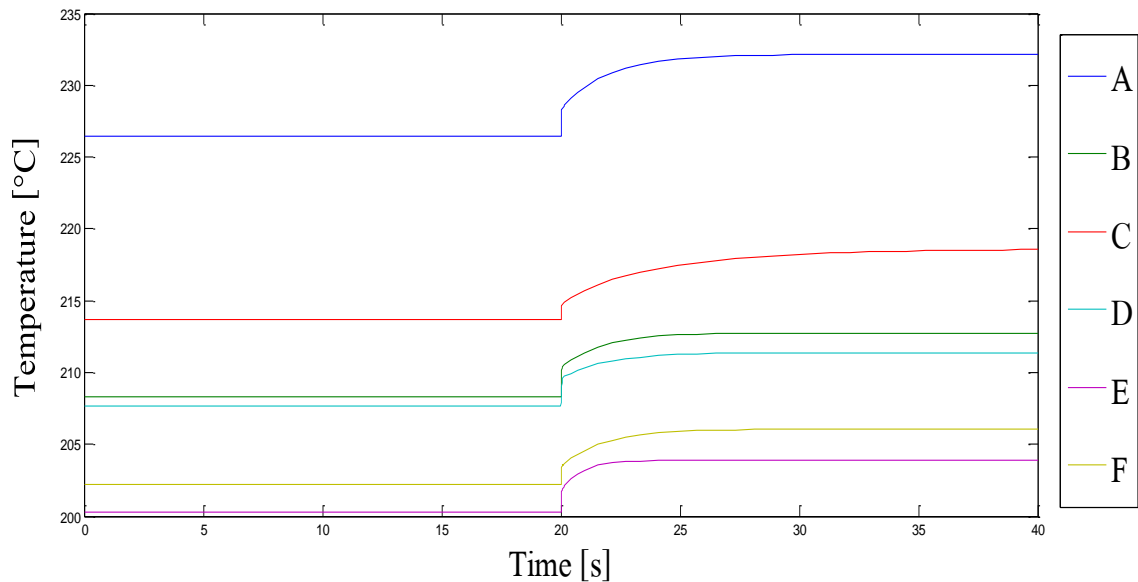


Fig. 33. Step change in the feed flowrate - Branches

The effect on the total outlet temperature is totally expected. The temperature grows directly due to the assumption of incompressibility and the direct heat balance and then stabilizes at a higher value due to the evolution of the temperature in the whole network.

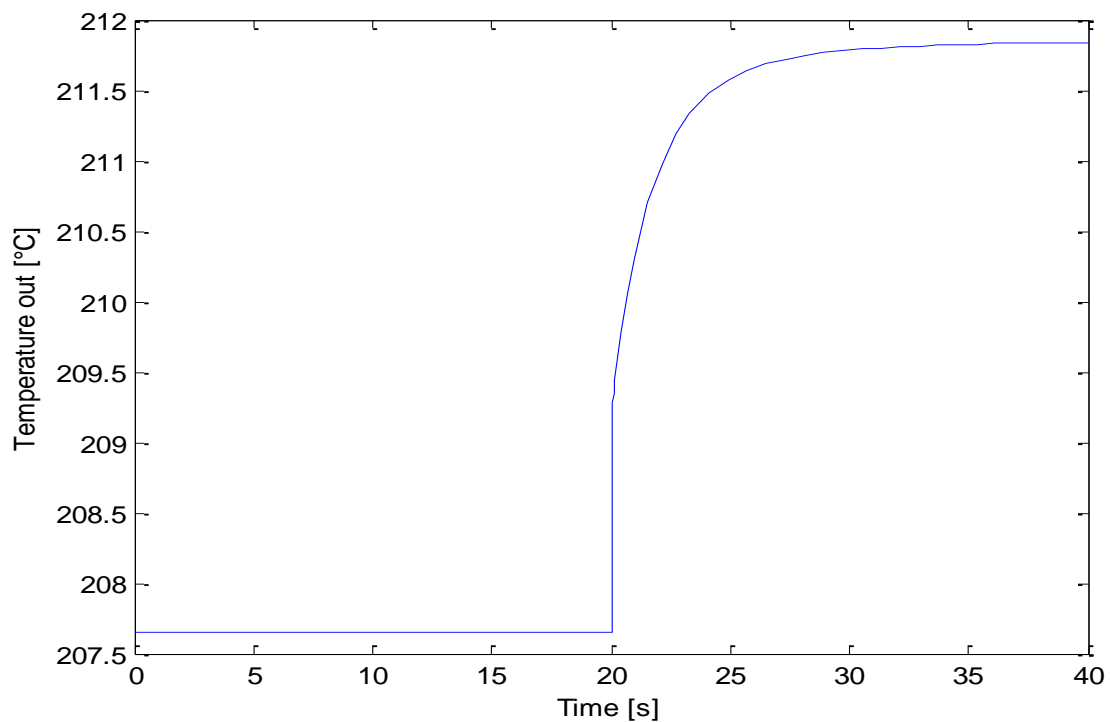


Fig. 34. Step change in the feed flowrate - Ttot

Step change in the split fractions (manipulated variable):

For each of the five split fractions used as degree of freedom (the split fraction of branch F is fixed by the mass balance), at time=20s, we simulate a step change of 0.1 (so the split fraction of branch F is reduced of 0.1).

This simulation shows once again the time-scale separation between the branches and the limitations of the model for very short-time scales. We observe that the split fraction of the fastest time-scale have a tendency to generate an overshoot since the branch F takes more time to bring additional heat in the final balance.

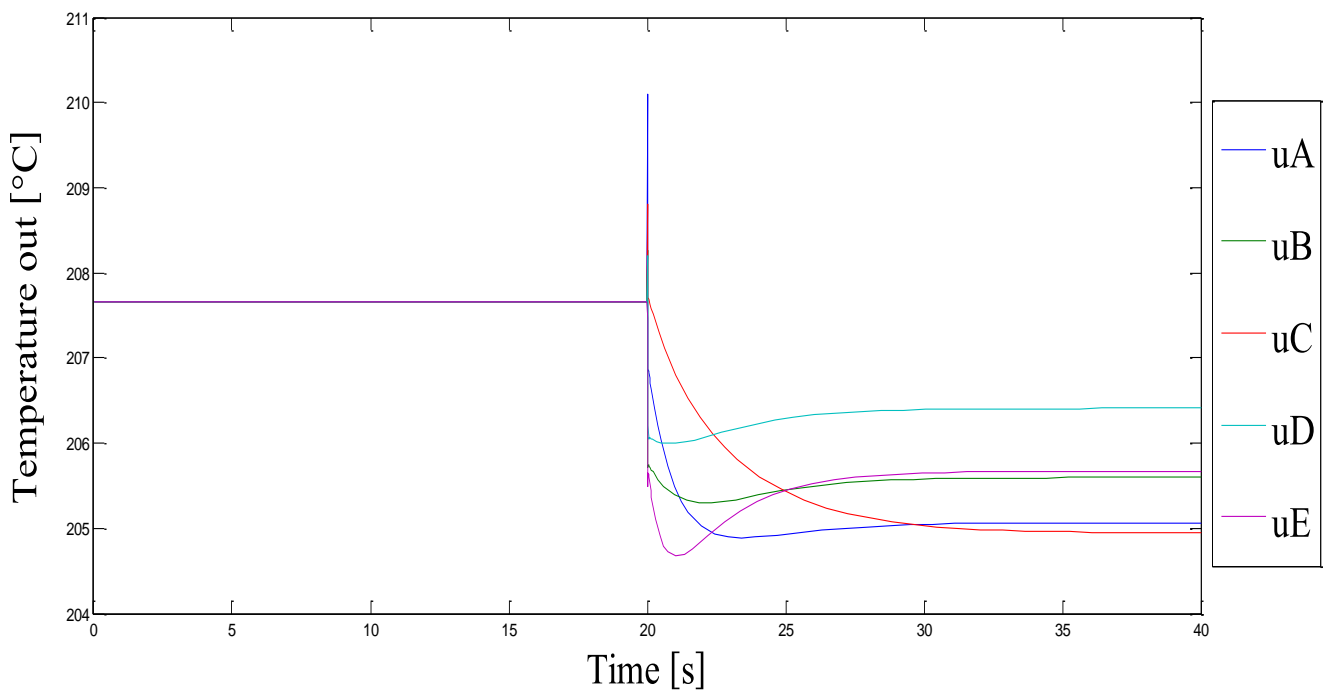


Fig. 35. Step change in the manipulated variables

6. CONCLUSION

In my previous education, I have been told that “*All models are wrong but some of them are useful.*” Indeed, modeling is always connected to objectives. My intention in this project was to work on the long term and build an accurate, robust and flexible heat exchanger network model that would be used for a large study of possible control configurations in the master thesis.

This objective is now considered to be reached since the current model is able to simulate the main characteristics of the dynamics of the crude unit heat exchanger network. The contribution of each unit or branch in the network is now clearly understood and the control implementation strategy can be built on this knowledge and this simulation quality.

Moreover, any additional information about flow configurations can be easily introduced due to the simplicity of the implementation in the designed m-files. Unfortunately, the assumption of incompressibility may bring problems such as algebraic loops on the very short-time scale. This eventual issue has to be reminded during the control implementation and properly assessed to not reduce the window of the study.

The main difficulty in this project was to estimate and exploit the necessary data for the computation of the state equations. A long time has been spent to guess variables and then, hopefully, the Statoil Mongstad staff provided almost every data that were requested. I would like to thank especially Thomas Løften and Kathrine Storsæter for the time they spent in this work. It was really exciting to fit the model with so fresh data in the best possible way.

The last simulation based on step change of the manipulated variables (cfr. Fig 35) shows an interesting time-scale separation between the branches. This may be a wonderful opportunity for introducing a simple control configuration based on PID controllers. The self-optimizing control method thus looks now even more promising for this crude unit heat exchanger network. This will be definitely the main control configuration to be deeply analyzed in the master thesis. Alternatives such as Model Predictive Control may also appear in the master thesis if a comparative study is of interest.

The heat exchanger network also needs to be considered as a whole and the effect of additional branches excluded of the study so far have to be re-introduced or, at least, re-examined.

Alexandre LERUTH,
MSc. in Chemical Engineering
ULg (Belgium) – NTNU (Norway)

Trondheim, the 16th of december 2011

7. BIBLIOGRAPHY

- Glemmestad, B., Skogestad, S., Gundersen, T., 1999. Optimal operation of heat exchanger networks. *Computers and Chemical Engineering* 23 (4-5), 509 – 522.
- Mathisen, K. W., Skogestad, S., Wolff, E. A., 1992. Bypass Selection for Control of Heat Exchanger Networks. *Computers and Chemical Engineering* 16 (Supplement 1), 263 – 272.
- Mathisen, K.W., Morari, M., Skogestad, S., Wolff, E. A., 1994. Optimal Operation of Heat Exchanger Networks Presented at Process Systems Engineering (PSE'94), Kyongju, Korea.
- Boyaci, C., Uzturk, D., Konukman, A. E. S., Akman, U., 1996. Dynamics and Optimal Control of Flexible Heat-Exchanger Networks. *Computers and Chemical Engineering* 20 (Supplement 2), 775 – 780.
- Lid, T., Strand, S., Skogestad, S., On-line optimization of a crude unit heat exchanger network, 2000
- Lid, T., Skogestad, S., 2001. Implementation issues for real-time optimization of a crude unit heat exchanger network. In: Gani, R., Jorgensen, S.B. (Eds.), *European Symposium on Computer Aided Process Engineering - 11, 34th European Symposium of the Working Party on Computer Aided Process Engineering*. Vol. 9 of *Computer Aided Chemical Engineering*. Elsevier, pp. 1041 – 1046.
- Chachuat, B., Srinivasan, B., Bonvin, D., 2009. Adaptation Strategies for Real-Time Optimization. *Computers and Chemical Engineering* 33 (10), 1557 – 1567.
- White, D. C., 1997. Online optimization: What, where and estimating ROI. *Hydro- carbon Processing* 76 (6), 4351.
- Skogestad, S., 2000. Plantwide control: the search for the self-optimizing control structure. *Journal of Process Control* 10.
- Jäschke, J., 2011. Invariants for Optimal Operation of Process Systems. Dissertation, Norwegian University of Science and Technology.
- Edvardsen, D.G., 2011, Optimal operation of Heat Exchanger Networks, master thesis, Norwegian University of Science and Technology.
- Sinnott, R., Towler, G., 2009. *Chemical Engineering Design*, 5th Edition. Elsevier Ltd., Oxford.
- Mathisen, K. W., Morari, M., Skogestad, S., 1994. Dynamic models for heat exchangers and heat exchanger networks. *Computers and Chemical Engineering* 18 (1), 459–463.
- www.mathworks.se/help/toolbox/simulink/
- I.A. Wiehe and R.J.Kennedy, 2000, *Energy and Fuels*, 14, 56-63
- Illustrations :
- www.wikipedia.org
- www.bbc.co.uk/schools/gcsebitesize/science/aqa/rocks/fuelsrev3.shtml

APPENDIX

DATA TREATMENT – TEMPERATURES AND FLOWS

Branch A :

$$\begin{aligned} \Delta H_{A,crude} &= \Delta H_{BC} \\ \Leftrightarrow m_A \int_{T_{in}}^{T_{A,out}} c_{p,crude} dT &= m_{BC} \int_{T_{BC,out}}^{T_{BC,in}} c_{p,BC} dT \\ \Leftrightarrow m_A \left[T_{A,out} \left(k_{1,crude} \frac{T_{A,out}}{2} + k_{2,crude} \right) - T_{in} \left(k_1 \frac{T_{in}}{2} + k_2 \right) \right] \\ &= m_{BC} \left[T_{BC,in} \left(k_{1,BC} \frac{T_{BC,in}}{2} + k_{2,BC} \right) - T_{BC,out} \left(k_{1,BC} \frac{T_{BC,out}}{2} + k_{2,BC} \right) \right] \end{aligned}$$

Since m_{BC} was missing in the asked data, this heat balance just serves to compute it. All others measured data are taken without any modification.

m_A [t/h]	T_{in} [°C]	$T_{A,out}$ [°C]	$T_{BC,in}$ [°C]	$T_{BC,out}$ [°C]
76,582	125,00	226,457	295,4453	167,59

$$\Rightarrow m_{BC} = 57,270 \text{ t/h}$$

Branch B :

$$\begin{aligned} \Delta H_{B,crude} &= \Delta H_{LGO} \\ \Leftrightarrow m_B \int_{T_{in}}^{T_{B,out}} c_{p,crude} dT &= m_{LGO,tot} \int_{T_{LGO,inter}}^{T_{LGO,in}} c_{p,LGO} dT + (m_{LGO,tot} - m_{LGO22}) \int_{T_{LGO,out}}^{T_{LGO,inter}} c_{p,LGO} dT \\ \Leftrightarrow m_B \left[T_{B,out} \left(k_{1,crude} \frac{T_{B,out}}{2} + k_{2,crude} \right) - T_{in} \left(k_1 \frac{T_{in}}{2} + k_2 \right) \right] \\ &= m_{LGO,tot} \left[T_{LGO,in} \left(k_{1,LGO} \frac{T_{LGO,in}}{2} + k_{2,LGO} \right) - T_{LGO,out} \left(k_{1,LGO} \frac{T_{LGO,out}}{2} + k_{2,LGO} \right) \right] \\ &\quad - m_{LGO22} \left[T_{LGO,inter} \left(k_{1,LGO} \frac{T_{LGO,inter}}{2} + k_{2,LGO} \right) - T_{LGO,out} \left(k_{1,LGO} \frac{T_{LGO,out}}{2} + k_{2,LGO} \right) \right] \end{aligned}$$

Since $m_{LGO,tot}$ and was missing in the asked data, this heat balance just serves to compute it. All others measured data are taken without any modification. The temperature $T_{B,inter}$ of the crude oil between the two exchangers can then be estimated by the heat balance on the first heat exchanger.

m_B [t/h]	T_{in} [°C]	$T_{B,out}$ [°C]	$T_{LGO,in}$ [°C]	$T_{LGO,inter}$ [°C]	$T_{LGO,out}$ [°C]	m_{LGO22} [t/h]
159,785	125,00	208,3047	267,820	235,285	179,117	39,930

$$\Rightarrow m_{LGO,tot} = 162,6105 \text{ t/h}$$

$$\Rightarrow T_{B,inter} = 172,202^\circ\text{C}$$

Branch C :

$$\begin{aligned}
\Delta H_{C,crude} &= \Delta H_{HGO} \\
\Leftrightarrow m_C \int_{T_{in}}^{T_{C,out}} c_{p,crude} dT &= m_{HGO} \int_{T_{HGO,out}}^{T_{HGO,in}} c_{p,HGO} dT \\
\Leftrightarrow m_C \left[T_{C,out} \left(k_{1,crude} \frac{T_{C,out}}{2} + k_{2,crude} \right) - T_{in} \left(k_1 \frac{T_{in}}{2} + k_2 \right) \right] \\
&= m_{HGO} \left[T_{HGO,in} \left(k_{1,HGO} \frac{T_{HGO,in}}{2} + k_{2,HGO} \right) - T_{HGO,out} \left(k_{1,HGO} \frac{T_{HGO,out}}{2} + k_{2,HGO} \right) \right]
\end{aligned}$$

Since m_{HGO} was missing in the asked data, this heat balance just serves to compute it. All others measured data are taken without any modification.

m_C [t/h]	T_{in} [°C]	$T_{C,out}$ [°C]	$T_{HGO,in}$ [°C]	$T_{HGO,out}$ [°C]
120,414	125,00	213,676	248,7734	141,254

$$\Rightarrow m_{HGO} = 97,075 \text{ t/h}$$

Branch D :

$$\begin{aligned}
\Delta H_{D1,crude} &= \Delta H_{MSR} \\
\Leftrightarrow m_D \int_{T_{in}}^{T_{D,inter}} c_{p,crude} dT &= m_{MSR} \int_{T_{MSR,out}}^{T_{MSR,in}} c_{p,MSR} dT \\
\Leftrightarrow m_D \left[T_{D,inter} \left(k_{1,crude} \frac{T_{D,inter}}{2} + k_{2,crude} \right) - T_{in} \left(k_1 \frac{T_{in}}{2} + k_2 \right) \right] \\
&= m_{MSR} \left[T_{MSR,in} \left(k_{1,MSR} \frac{T_{MSR,in}}{2} + k_{2,MSR} \right) - T_{MSR,out} \left(k_{1,MSR} \frac{T_{MSR,out}}{2} + k_{2,MSR} \right) \right] \\
\Delta H_{D2,crude} &= \Delta H_{BSR} \\
\Leftrightarrow m_D \int_{T_{D,inter}}^{T_{D,out}} c_{p,crude} dT &= m_{BSR} \int_{T_{BSR,out}}^{T_{BSR,in}} c_{p,BSR} dT \\
\Leftrightarrow m_D \left[T_{D,out} \left(k_{1,crude} \frac{T_{D,out}}{2} + k_{2,crude} \right) - T_{D,inter} \left(k_{1,crude} \frac{T_{D,inter}}{2} + k_{2,crude} \right) \right] \\
&= m_{BSR} \left[T_{BSR,in} \left(k_{1,BSR} \frac{T_{BSR,in}}{2} + k_{2,BSR} \right) - T_{BSR,out} \left(k_{1,BSR} \frac{T_{BSR,out}}{2} + k_{2,BSR} \right) \right]
\end{aligned}$$

In these balances, we hopefully have all the data. But the equations do not match perfectly and reconciliation is needed.

m_D [t/h]	T_{in} [°C]	$T_{D,inter}$ [°C]	$T_{MSR,in}$ [°C]	$T_{MSR,out}$ [°C]	m_{MSR} [t/h]
177,516	125,00	181,06	222,81	176,58	213,63

$T_{D,out}$ [°C]	$T_{BSR,in}$ [°C]	$T_{BSR,out}$ [°C]	m_{BSR} [t/h]
207,00	268,617	243,906	174,090

For the given values, we observe that :

$$\Delta H_{D1} \ll \Delta H_{MSR}$$

$$\Delta H_{D2} < \Delta H_{BSR}$$

So we introduce two reconciliation factors $\varepsilon > 0$ and $\gamma > 0$ in order to enhance the overall heat transfer and redistribute as little as possible the heat exchange between the units :

$$m_{r,D} = m_D(1 + 2\varepsilon) \quad ; \quad T_{r,D,out} = T_{D,out}(1 + \varepsilon)$$

$$T_{r,MSR,in} = T_{MSR,in}(1 - \varepsilon) \quad ; \quad T_{r,MSR,out} = T_{MSR,out}(1 + \varepsilon)$$

$$m_{r,MSR} = m_{MSR}(1 - 2\varepsilon) \quad ; \quad m_{r,BSR} = m_{BSR}(1 - 2\varepsilon)$$

$$T_{r,BSR,in} = T_{BSR,in}(1 - \varepsilon) \quad ; \quad T_{r,BSR,out} = T_{BSR,out}(1 + \varepsilon)$$

$$T_{r,D,inter} = T_{D,inter}(1 + \gamma)$$

And we solve the previous heat balances with the new temperatures and flowrates in order to find:

$$\Rightarrow \varepsilon = 0,003296 \quad ; \quad \gamma = 0,01389$$

The corrected values are:

$m_{r,D}$ [t/h]	T_{in} [°C]	$T_{r,D,inter}$ [°C]	$T_{r,MSR,in}$ [°C]	$T_{r,MSR,out}$ [°C]	$m_{r,MSR}$ [t/h]
178,686	125,00	181,06	222,078	177,164	212,225

$T_{r,D,out}$ [°C]	$T_{r,BSR,in}$ [°C]	$T_{r,BSR,out}$ [°C]	$m_{r,BSR}$ [t/h]
207,68	267,632	244,710	172,942

Branch E :

$$\Delta H_{E,crude} = \Delta H_{KERO}$$

$$\Leftrightarrow m_E \int_{T_{in}}^{T_{E,out}} c_{p,crude} dT = m_{KERO} \int_{T_{KERO,out}}^{T_{KERO,in}} c_{p,KERO} dT$$

$$\Leftrightarrow m_E \left[T_{E,out} \left(k_{1,crude} \frac{T_{E,out}}{2} + k_{2,crude} \right) - T_{in} \left(k_1 \frac{T_{in}}{2} + k_2 \right) \right]$$

$$= m_{KERO} \left[T_{KERO,in} \left(k_{1,BC} \frac{T_{KERO,in}}{2} + k_{2,BC} \right) - T_{KERO,out} \left(k_{1,BC} \frac{T_{KERO,out}}{2} + k_{2,KERO} \right) \right]$$

In this balance, we hopefully have all the data. But the equation does not match perfectly so reconciliation is needed.

m_E [t/h]	T_{in} [°C]	$T_{E,out}$ [°C]	$T_{KERO,in}$ [°C]	$T_{KERO,out}$ [°C]	m_{KERO} [t/h]
126,092	125,00	200,070	241,836	180,551	142,441

For the given values, we observe that :

$$\Delta H_{E,crude} < \Delta H_{KERO}$$

So we introduce the reconciliation factors $\varepsilon > 0$ such that :

$$m_{r,E} = m_E(1 + 2\varepsilon)$$

$$T_{r,E,out} = T_{E,out}(1 + \varepsilon)$$

$$T_{r,KERO,in} = T_{KERO,in}(1 - \varepsilon)$$

$$T_{r,KERO,out} = T_{KERO,out}(1 + \varepsilon)$$

$$m_{r,KERO} = m_{KERO}(1 - 2\varepsilon)$$

And we solve the previous heat balances with the new temperatures and flowrates in order to find ε :

$$\Rightarrow \varepsilon = 0,00103$$

The corrected values are:

$m_{r,E}$ [t/h]	T_{in} [°C]	$T_{r,E,out}$ [°C]	$T_{r,KERO,in}$ [°C]	$T_{r,KERO,out}$ [°C]	$m_{r,KERO}$ [t/h]
126,351	125,00	200,276	241,587	180,7366	142,1482

Branch F : cfr. report.

Global balances on the crude oil :

Mass balance :

$$m_{tot} = m_A + m_B + m_C + m_{r,D} + m_{r,E} + m_{r,F}$$

Since m_{tot} was not part of the asked data, this mass balance is used to estimate it.

$$\Rightarrow m_{tot} = 915,4227 \text{ t/h}$$

Heat balance (cfr. 2.3.1. (*)) :

$$T_{tot} = \frac{-m_{tot}k_2 + \sqrt{m_{tot}^2k_2^2 + 2m_{tot}k_1 \sum_{streams} m_i T_i (k_1 \frac{T_i}{2} + k_2)}}{m_{tot}k_1}$$

In order to simplify the reconciliation procedure, T_{tot} is calculated by this expression.

$$T_{tot} = 207,6543^\circ\text{C}$$

For information, the measured T_{tot} was $206,6836^\circ\text{C}$. The reconciliation forces it to a 0,47% adjustment.

MATLAB © CODE

Please refer to the Fig.11. for Simulink user interface.

Intro.m

```
%Intro - load initial values, 23-10-2011 18h10 36''+reconciliation
```

```
%CRUDE OIL
```

```
T0=125; %°C
```

```
m0A=76.582/3.6; %t/h -> kg/s
```

```
m0B=159.785/3.6;
```

```
m0C=120.414/3.6;
```

```
m0D=178.686/3.6;
```

```
m0E=126.351/3.6;
```

```
m0F=253.604/3.6;
```

```
m0=m0A+m0B+m0C+m0D+m0E+m0F;
```

```
%Branch A
```

```
uA=m0A/m0;
```

```
ThinA=295.4453;
```

```
cp_hA=k1A*ThinA+k2A;
```

```
mhA=57.270/3.6;
```

```
%Branch B
```

```
uB=m0B/m0;
```

```
ThinB=267.82;
```

```
cp_hB=k1B*ThinB+k2B;
```

```
mhB=162.6105/3.6;
```

```
LGO22=39.92969/3.6;
```

```
%Branch C
```

```
uC=m0C/m0;
```

```
ThinC=248.7734;
```

```
cp_hC=k1C*ThinC+k2C;
```

```
mhC=97.075/3.6;
```

```
%Branch D
```

```
uD=m0D/m0;
```

```
ThinD1=222.078;
```

```
cp_hD1 =k1D1*ThinD1+k2D1;
```

```
mhD1=212.225/3.6;
```

```
ThinD2=267.632;
```

```
cp_hD2=k1D2*ThinD2+k2D2;
```

```
mhD2=172.942/3.6;
```

```
%Branch E
```

```
uE=m0E/m0;
```

```
ThinE=241.587;
```

```
cp_hE=k1E*ThinE+k2E;
```

```
mhE=142.1482/3.6;
```

```
% Branch F
```

```
ThinF1=193.843;
```

```
cp_hF1=k1F1*ThinF1+k2F1;
```

```
mhF1=72.125/3.6;
```

```
ThinF2=244.379;
```

```
cp_hF2=k1F2*ThinF2+k2F2;
```

```
mhF2=(113.081+110.689)/3.6;
```


Heat exchanger	S-function m-file [hexchXX.m]	Indexch [Y]
A	hexch3.m	4
B1	hexch41.m	5
B2	hexch42.m	6
C1	hexch51.m	7
C2	hexch52.m	8
D1	hexch61.m	9
D2	hexch62.m	10
E	hexch2.m	3
F1	hexch11.m	1
F2	hexch12.m	2

hexchXX.m

```
function [sys,x0] = hexchXX(t,x,u,flag)
Indexch=Y;
N = 10;
%
% Simulink interface, exchanger consisting of three series of lump systems :
% Hot side, Wall, Cold side
%
% t - time in [s].
% X - State, the first N states are hot temperatures,
% the second N states are wall temperatures,
% the last N states are cold temperatures.
% Inputs:
% U(1) - Inlet hot temperature
% U(2) - Inlet hot mass flow
% U(3) - Inlet cold temperature
% U(4) - Inlet cold mass flow
%
% Outputs: When flag is 0 sys contains sizes and x0 contains initial condition.
% When flag is 1, sys contains the state derivatives.
% When flag is 3 sys contains outputs;
% y(1) - Outlet hot temperature
% y(2) - Outlet hot mass flow
% y(3) - Outlet cold temperature
% y(4) - Outlet cold mass flow

if abs(flag) == 1
    % Return state derivatives.
    sys = mathisen(t,x,u,N,Indexch);

elseif abs(flag) == 3 % Return system outputs.

    sys(1,1) = x(N); % Outlet hot temperature
    sys(2,1) = x(3*N); % Outlet cold temperature

elseif flag == 0 % Initalize the system

    x0 = steadyvar(Indexch,N);
    sys = [3*N,0,2,4,0,0]; %number of states, outputs, inputs
else
    sys = [];
end
end
```

Steadyvar.m

```

function [x0]=steadyvar(Indexch,N)

if N==10
    if Indexch==1
        x0 =
[188.821189347093,184.064371788945,179.563318018329,175.308633179628,171.29
0812489110,167.500294502864,163.927511482946,160.562936414220,157.397126327
431,154.420761684806,140.172731397991,142.154641626783,144.264464582883,146
.508736188978,148.894114621792,151.427356930430,154.115292927563,156.964796
452930,159.982754170073,163.176032121504,125.924701111176,126.912156926135,
127.965992751546,129.089960895010,130.287934740721,131.563901371750,132.921
952675497,134.366274887530,135.901136551200,137.530874895915];
    elseif Indexch==2
        x0 =
[236.542548541706,228.815616964684,221.198032390770,213.689580881418,206.29
0007301292,198.999015257865,191.816267118445,184.741384105692,177.773946472
384,170.913493755841,157.249460057196,163.752618499965,170.355113388803,177
.057345790208,183.859679514479,190.762440921215,197.765918796286,204.870364
299790,212.075990984196,219.382974881577,143.585426358551,149.731290527546,
155.968842671915,162.298424461971,168.720343771094,175.234874541139,181.842
256711153,188.542696208809,195.336365003707,202.223401221448];
    elseif Indexch==3
        x0 =
[236.373709983483,230.982947919496,225.404371406103,219.626676693288,213.63
7471140236,207.423123259489,200.968585365540,194.257182486298,187.270359391
236,179.987375167395,156.991644174346,164.935106621032,172.549299704039,179
.857723120800,186.881362944870,193.639046937299,200.147738284952,206.422780
261969,212.478101429496,218.326388846243,133.995913181297,142.599853850828,
150.841416921781,158.746860876059,166.339602630251,173.640622734361,180.668
799876615,187.441189117834,193.973254939495,200.279067709003];
    elseif Indexch==4
        x0 =
[280.105847433722,265.270155263516,250.942430745715,237.125749239346,223.82
1986946348,211.031764266429,198.754401507834,186.987888481732,175.728869220
856,164.972642710383,149.132063850481,158.857418773925,169.038275704907,179
.679070574804,190.783037781288,202.352208176369,214.387421221824,226.888350
549594,239.853541855826,253.280461805001,133.291484990579,141.985968326993,
151.088662928083,160.603739641774,170.534311296148,180.882429406391,191.649
093204303,202.834270353472,214.436928448136,226.455076176281];
    elseif Indexch==5
        x0 =
[229.238047599607,223.405389839306,217.617771012971,211.875806451550,206.18
0112343944,200.531305193487,194.930001245251,189.376815883875,183.872363001
763,178.417254337545,153.974046795688,158.988798857398,164.049753115224,169
.156353847588,174.308042720652,179.504259408939,184.744442189478,190.028028
509423,195.354455527254,200.723160627711,129.530839253832,134.105234713033,
138.722690346573,143.382706449925,148.084780247816,152.828406473935,157.613
077927406,162.438286005875,167.303521215201,172.208273655815];
    elseif Indexch==6
        x0 =
[264.692200523635,261.534750811916,258.346932040025,255.127997102887,251.87
7169059616,248.593639467202,245.276566593665,241.925073499929,238.538245978
479,235.115130335586,205.559583175918,209.146689979542,212.693956598144,216
.202467723492,219.673261762548,223.107333507530,226.505636613024,229.869085
896813,233.198559479426,236.494900775942,176.004036016250,179.755133980605,
183.462839696359,187.128368853319,190.752884057893,194.337497955443,197.883
276123161,201.391239753600,204.862368146937,208.297601028248];
    
```

```

elseif Indexch==7
    x0 =
[182.272292850455,176.843076585376,171.615603404262,166.585564886212,161.74
8520841898,157.099915197607,152.635091998085,148.349311397130,144.237765510
690,140.295594014675,134.247219694495,137.887419478860,141.684082346227,145
.641708670084,149.764751941108,154.057604478357,158.524582903218,163.169913
483247,167.997717459778,173.011996478471,128.198845374316,131.537073447029,
135.018853295324,138.648325342083,142.429588684609,146.366688114815,150.463
600920224,154.724223562233,159.152358334180,163.751700106486];
elseif Indexch==8
    x0 =
[241.856531929863,235.128785287557,228.588722295451,222.234732231908,216.06
5035112840,210.077686121872,204.270580760658,198.641460683738,193.187920175
634,187.907413221824,177.987752554880,182.858447958401,187.889069413236,193
.081998862273,198.439490562996,203.963664534985,209.656500542987,215.519832
659953,221.555344449053,227.764564797621,168.068091887936,172.528975741168,
177.136678142734,181.893416963889,186.801295004120,191.862293957131,197.078
268854066,202.450943024455,207.981903610549,213.672597665378];
elseif Indexch==9
    x0 =
[218.366748198809,214.538364071445,210.586942906941,206.506116787787,202.28
9003090486,197.928145444251,193.415445759144,188.742085623184,183.898434972
747,178.873945435695,155.230831495342,160.909458721864,166.378284382907,171
.649914287006,176.735827548507,181.646510395674,186.391570377922,190.979834
353955,195.419432976257,199.717873867675,131.587717554989,137.920482470982,
144.014483142630,149.884382814868,155.543509652762,161.004017700862,166.277
023968057,171.372725800970,176.300501881070,181.068999536542];
else
    x0 =
[265.138344161081,262.631942880567,260.112580684217,257.580036311805,255.03
4082501553,252.474485764237,249.901006146347,247.313396981656,244.711404630
501,242.094768206030,212.950072597160,215.618047401056,218.270330066318,220
.907208302648,223.528961619759,226.135861644356,228.728172421704,231.306150
702672,233.870046217110,236.420101934322,183.805376988289,186.524690171610,
189.227263150980,191.913410458950,194.583437475281,197.237640787159,199.876
308531602,202.499720721127,205.108149553653,207.701859707563];
end

else
HXinitA = [295.4453,167.5898,125,226.457];

HXinitB1 = [235.2852,179.1172,125,172.202];
HXinitB2 = [267.8203,235.2852,172.202,208.3047];

HXinitC1 = [176.75,141.2539,125,154.112];
HXinitC2 = [248.7734,176.75,154.112,213.6758];

HXinitD1 = [222.0782,177.164,125,183.5775];
HXinitD2 = [267.7319,244.7101,183.5775,207.6822];

HXinitE = [241.587,180.7366,125,200.2762];

HXinitF1 = [193.8429,154.6104,125,137.535];
HXinitF2 = [244.3789,171.7525,137.535,202.2197];

```

```

if Indexch==1
    exch=HXinitF1;
elseif Indexch==2
    exch=HXinitF2;
elseif Indexch==3
    exch=HXinitE;
elseif Indexch==4
    exch=HXinitA;
elseif Indexch==5
    exch=HXinitB1;
elseif Indexch==6
    exch=HXinitB2;
elseif Indexch==7
    exch=HXinitC1;
elseif Indexch==8
    exch=HXinitC2;
elseif Indexch==9
    exch=HXinitD1;
else
    exch=HXinitD2;
end

hotstep=(exch(2)-exch(1))/(N-1);
coldstep=(exch(4)-exch(3))/(N-1);
Thot0=exch(1):hotstep:exch(2);
Tcold0=exch(3):coldstep:exch(4);
Twall0=(Thot0+Tcold0)/2;
x0=[Thot0,Twall0,Tcold0];

end

```

mathisen.m

```

function xprime=mathisen(t,X,U,N,Indexch)

Thot=X(1:N);
Twall=X(N+1:2*N);
Tcold=X(2*N+1:3*N);
Thin=U(1);
Mhin=U(2);
Tcin=U(3);
Mcin=U(4);

[Ai,Vi,V_w,rho_w,cp_w]=data(Indexch,N);

cp_c = 1000*(2.0*2.1297e-003)*Tcold + 1000*1.7895; %CRUDE
rho_c = 0.73596;

if Indexch==1
    cp_h = 1000*(2.0*2.4663e-003*Thot + 1.7791); %HNA
    rho_h = 0.663196;
    h_c = 1381; %W/(m^2K)
    h_h = h_c;
elseif Indexch==2
    cp_h = 1000*(2.0*1.8189e-003*Thot + 1.7798); %RES
    rho_h = 0.83286;
    h_c = 1348; %W/(m^2K)
    h_h = h_c;

```

```

elseif Indexch==3
    cp_h = 1000*(2.0*2.2148e-003*Thot + 1.7941); %KERO
    rho_h = 0.682727;
    h_c = 1976; %W/(m^2K)
    h_h = h_c;
elseif Indexch==4
    cp_h = 1000*(2.0*1.9783e-003*Thot + 1.7771); %BC
    rho_h = 0.736938; %BC like BSR
    h_c = 1902; %W/(m^2K)
    h_h = h_c;
elseif Indexch==5 | Indexch==6
    cp_h = 1000*(2.0*2.0636e-003*Thot + 1.7963); %LGO
    rho_h = 0.718929;
    if Indexch==5
        h_c = 1189; %W/(m^2K)
        h_h = h_c;
    else
        h_c = 713; %W/(m^2K)
        h_h = h_c;
    end
elseif Indexch==7 | Indexch==8
    cp_h = 1000*(2.0*1.9901e-003*Thot + 1.7924); %HGO
    rho_h = 0.788892;
    h_c = 1565; %W/(m^2K)
    h_h = h_c;
elseif Indexch==9
    cp_h = 1000*(2.0*2.2292e-003*Thot + 1.7967); %MSR
    rho_h = 0.690296; %MPA
    h_c = 1250; %W/(m^2K)
    h_h = h_c;
else
    cp_h = 1000*(2.0*2.0483e-003*Thot + 1.7877); %BSR
    rho_h = 0.719023; %BPA
    h_c = 383; %W/(m^2K)
    h_h = h_c;
end

w_h=Mhin*cp_h;
w_c=McIn*cp_c;

dThotdt(1)=(Thin-Thot(1)-((h_h*Ai)/(w_h(1)*N))*(Thot(1)-
Twall(N)))*((Mhin*N)/(rho_h*Vi));
dTwalldt(1)=(h_h*(Thot(N)-Twall(1))-h_c*(Twall(1)-
Tcold(1)))*(Ai/(rho_w*cp_w*V_w));
dTcolddt(1)=(Tcin-Tcold(1)-((h_c*Ai)/(w_c(1)*N))*(Tcold(1)-
Twall(1)))*((McIn*N)/(rho_c*Vi));

for i=2:N
    j=N-i+1;
    dThotdt(i)=(Thot(i-1)-Thot(i)-((h_h*Ai)/(w_h(i)*N))*(Thot(i)-
Twall(j)))*((Mhin*N)/(rho_h*Vi));
end
for j=2:N
    i=N-j+1;
    dTwalldt(j)=(h_h*(Thot(i)-Twall(j))-h_c*(Twall(j)-
Tcold(j)))*(Ai/(rho_w*cp_w*V_w));
dTcolddt(j)=(Tcold(j-1)-Tcold(j)-((h_c*Ai)/(w_c(j)*N))*(Tcold(j)-
Twall(j)))*((McIn*N)/(rho_c*Vi));
end

xprime=[dThotdt,dTwalldt,dTcolddt];

```

data.m

```
function [Ai,Vi,V_w,rho_w,cp_w]=data(Indexch,N)

A      = [77,556,164,138,162,203,264,233,260,313];
Ai     = A(Indexch);
Vshell = [0.67,5.03,1.2,1.7,1.287,1.718,2.38,2.049,1.86,2.61];
Vbundle= [0.45,2.72,0.7,0.8,0.712,0.96,1.4,1.151,1.25,1.88];
Vtot   = (Vshell+Vbundle); %to be modified when idea of flow configuration
Vi     = Vtot(Indexch)/2;

% Wall
cp_w   = 460; %P, Wall heat capacity [J/kg*K]
rho_w  = 7800; %P, Wall density [kg/m3]

mass_w = [1580,11130,3060,3760,3080,4070,5480,4830,4910,6400]; %Guess 4040
V_w    = mass_w(Indexch)/(rho_w*N); % Wall volumes [m3]
```