Modelling and Simulation of a Supermarket Refrigeration System

Special Course

s172040, Zoltán Márk Pintér
Supervisor: Dimitrios Papageorgiou

Technical University of Denmark
Department of Electrical Engineering
March 5, 2019

Contents

Chapter 1

Introduction

Motivation

The motivation of this paper is to model a supermarket refrigeration system for designing controllers. These controllers use measurements, constraints for outputs, lifetime and performance considerations in order to control the control output signals. This approach needs a good fidelity for giving a nonlinear model with the most characteristic dynamics of the investigation thermodynamic system, with the correct steady states. On the other hand, there is no need for high fidelity distributed models, which would be burdening for deriving and computation exhaustive for simulations. It is, however, significant to give a good range of parameters from industrial data, in order to make reliable parameter or state estimations for stable and efficient feedback, or fault diagnosis. The architecture of the prepared Matlab code is object oriented and modular.

Background Literature

The general background knowledge for investigating thermodynamic systems and heat transfer can be found in the book of *Cengel*, [1], and this basically covers everything that is necessary to understand the physical phenomena below. The different approaches of fluid dynamics and thermodynamics are often difficult to adjust; a book that connects them well, is written by *Lajos* [3]. The dimensioning of the hydraulic resistances is found in the book of *Granryd* [2]. The structures of state-of-the-art supermarket refrigeration systems are taken from the company *Danfoss*, RAC department. The implementation uses some of the object oriented concepts, which are described in the book of *Rumbaugh* [4].

Refrigeration systems

A standard refrigeration cycle (see Figure 1.2, left side) has two pressure levels, since the temperature increases with the pressure, which results in more effective heat transfer for the same mass flow rate - as the heat transfer increases by the temperature differences. The compressors and the valves separate the pressure levels, both of them control the pressure on their inlet side. On each pressure levels, there are heat exchangers, which are called condensor and evaporator. Condensing and evaporation is to leverage that there is no change in the temperature along the varying enthalpy, ensuring more effective heat transfer. Note, however, that our system works with carbon-dioxide refrigerant, which needs to go over the critical point (see Figure 1.1, where the critical point is the top of the saturation curve) for achieving a good heat transfer. In the right side of Figure 1.2 it can also be seen that a buffer unit called receiver is used in practice, and this introduces a new, intermediate pressure level. This is not only a mass buffer, but it separates the liquid from the gas from the incoming two-phase fluid, since the liquid provides better heat transfer characteristics,

and it is easier to pipe. Another notable consideration is that the compressor's lifetime decreases significantly, if it takes liquid on its suction side.

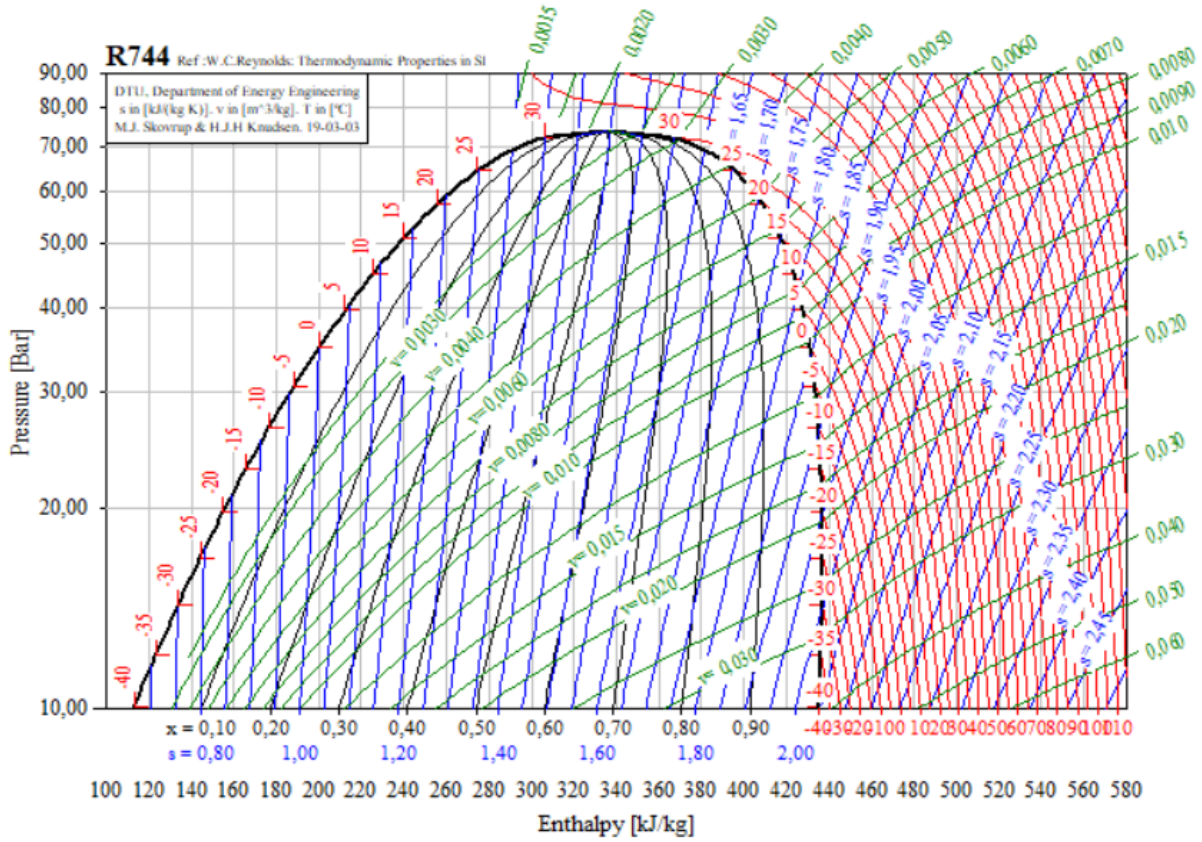

Figure 1.1: The ph (pressure-enthalpy) diagram of CO2 (carbon-dioxide)

One more modification to this model is if the low and middle temperature pressure levels are introduced (see Figure 1.2, down). This is again due to the temperature-pressure correlation, which gives the possibility to control different temperature levels, like the ones in the cooler and the freezer.

There are several timescales and dynamics appearing in a refrigeration system, which range from sound dynamics through mechanics and fluid dynamics to thermodynamics. Even on the thermodynamic level, the high pressure and receiver pressure level dynamics are considered as being independent from the pressure levels of the evaporators. Evaporator control is more distributed and therefore the dynamics are faster, while the gas cooler becomes the slowest element of the system. This reasons a distributed model of the latter component, to have a better idea about how the pressure level and the temperature distribution of it changes along its tubes. However, the lower pressure levels were considered as well, in order to model the possible dynamics due to the refrigerant loop (for example, as it turns out, the model might integrate energy in case of insufficient modelling).

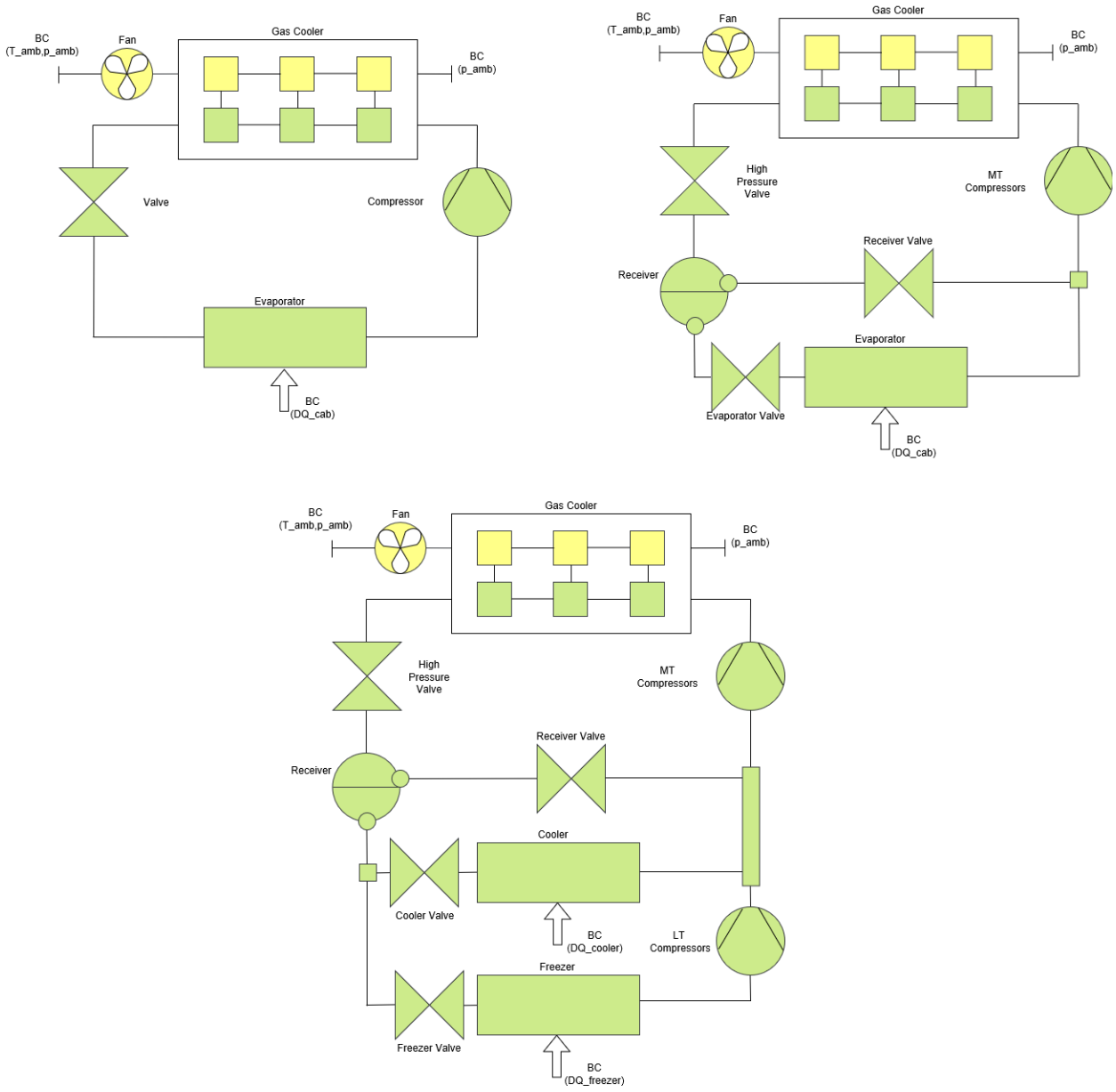

Figure 1.2: The refrigeration cycle. Left side: standard setup. Right side: Setup with receiver. Down: Two-level setup. *Notations: cab* $\sim$ *cabinet, amb* $\sim$ *ambient,* $DQ \sim$ heat flow rate

Notes

In Section 2, the notations, and in Section 3, the most used differential equations are introduced as general knowledge. The dynamics of the model are described in Section 4, while the algebraic constraints needed to integrate the states are stated in Section 5. The deduction of equations and the most important assumptions are stated in Section 6. In Section 7, the object oriented and modular code architecture is discussed. The results are depicted in Section 8 and are they compared to the coding and solving strategies with a popular modelling software, TIL Suite, in Section 9. The conclusions and the further perspectives are described in Section 10, and some of the failed ideas are discussed in the appendix, Section 11.

Alongside this project, another version of the refrigeration system has been designed in another programming environment, Modelica, with the help of a library, TIL Suite, and solvers provided

by Dymola, and the results are going to be compared. The parameters were taken from industrial data, which is partly confidential, therefore they are not explicitly stated in this paper. However, where verification is needed, sufficient information will be provided to interpret the plot results.

Chapter 2

Notations

The generally used notations can be found in Table 2.1.

Notation	Unit	Description
p	Pa	Pressure
h	J kg^{-1}	Specific enthalpy
ρ	kg m^{-3}	Density
T	K	Absolute temperature
$\dot{m}$	kg s^{-1}	Mass flow rate
$\dot{V}$	m^3s^{-1}	Volume flow rate
σ	W K^{-1}	Thermal conduction
$\dot{Q}$	W	Heat flow rate
W_t	J s^{-1}	Technical work
τ	s	Time constant (Slowness of an actuator)
CR	-	Capacity ratio (Utilization of an actuator)
f	Hz	Frequency

Table 2.1: Table of notations

Note that the dependencies of partial differentials are not expressed in the documentation. However, every partial differential depends on two states - which describe the equilibrium entirely -, for example pressure and enthalpy. This means that $\frac{\partial z}{\partial x}|_y(x, y)$ will be denoted as $\frac{\partial z}{\partial x}|_y$.

Chapter 3

Differential equations of thermodynamic systems

There are four main differential equations, which are used to deduce the equations of the system. These are rule for partial derivatives in mathematics, the mass balance equation, the energy balance

equation and the low pass filter equation.

When we intend to convert time derivatives of states to time derivatives of other states, it is enough to consider the derivatives of two other states. This can be reasoned by Gibb's rule, which states

$$f + N = K + 2, \tag{3.1}$$

where f is the degrees of freedom, N is the number of phases (that is either 1 or 2), and K is the number of components (that is 1). Therefore, we need at maximum two states to determine a third one, and in equilibrium it will be true for small changes in states that

$$\mathrm{d}z = \frac{\partial z}{\partial x}|_y \mathrm{d}x + \frac{\partial z}{\partial y}|_x \mathrm{d}y. \tag{3.2}$$

The variables x, y, z are states of thermodynamic systems. Dividing the equation with infinitesimally small time $\mathrm{d}t$, we can determine the necessary time derivatives, when two other ones are provided. Note that the equation is true for a point in the ph diagram, which reflects equilibrium (since the partial derivatives are changing with the states). This will not hold at any time, since we investigate the dynamical behaviour of thermodynamic systems; however, given that this is the best estimation for the partial derivatives, and that they are changing slowly, we can safely assume that the results will be reliable.

The mass balance equation of open systems [1] can be stated for a given control volume. It shows that the density in the control volume is increased by inlet mass flow rates and decreased by outlet mass flow rates.

$$\dot{\rho} V = \sum_j \dot{m}_{in,j} - \sum_k \dot{m}_{out,k} \tag{3.3}$$

The energy balance equation of open systems [1] can be stated for a given control volume. It shows that the energy in the control volume is increased by the inlet enthalpy flow rates (first sum), decreased by the outlet enthalpy flow rates (second sum) and increased by inlet heat flow rate and technical work W_t (if it makes work *on* the fluid in the control volume). The equation is used with zero on the left side, when concentrated elements are considered, since for them no accumulation is assumed. Note that potential field (like gravity) and the kinetic energy of the fluid are not considered.

$$\dot{u}\rho V = \sum_j \dot{m}_{in,j} h_{in,j} - \sum_k \dot{m}_{out,k} h_{out,k} + \dot{Q} + \dot{W}_t \tag{3.4}$$

The low pass filter is the simplest way of creating dynamics for a concentrated element. It is used for realizing actuator dynamics and for damping interactions between states. Given that x is the state and u is the input, the equation states that

$$\dot{x} = \frac{1}{\tau}(-x + u). \tag{3.5}$$

Chapter 4

Model description

The refrigeration system investigated is a system, which is externally excited by four boundary conditions and four actuators. The system is made up of a gas cooler (counterflow heat exchanger), through which a fan pushes air, a high pressure valve, which is to keep the pressure (therefore the temperature) in the gas cooler high, the receiver, which is a mass buffer in the system, and separates two-phase fluid to gas and liquid, a receiver valve, which is to keep the pressure level of the receiver at higher level than the evaporator pressures (so that mass flow can be controlled through the latter), compressors, which are to keep the cooler evaporator pressure level at the correct level. In order to model the pressure of the suction side of the compressor, the volume of cooler was semi-considered, but the freezer evaporator was modelled as a boundary condition entirely.

The boundary conditions are the inlet air temperature and (standard air) pressure, the outlet (standard) air pressure, the heat flow rate into the evaporators from the supermarket, and the outlet enthalpies of them, assuming that the controllers setting the superheat of them are working perfectly. Therefore the cooler evaporator volume - being thermally insulated - receives the enthalpy that it is supposed to output.

The gas cooler is divided into cells with equal volumes. This is due to the fact that the heat transfer is strongly nonlinear, and has several characteristic, because of which continuous or well-discretized model is to be used. One example is that the heat transfer is practically saturated (the maximum is achieved) at the so-called pitch temperature difference, which can occur at any points along the tube. The heat exchange is modelled as counter flow, since although geometrically cross flow is happening, but the same fluid is lead through the serial dimension more times, resulting in gradually changing temperature for both media. For the dimensions of the gas cooler, see Figure 4.2 (the picture has been taken from the documentation of TIL Suite library (click)).

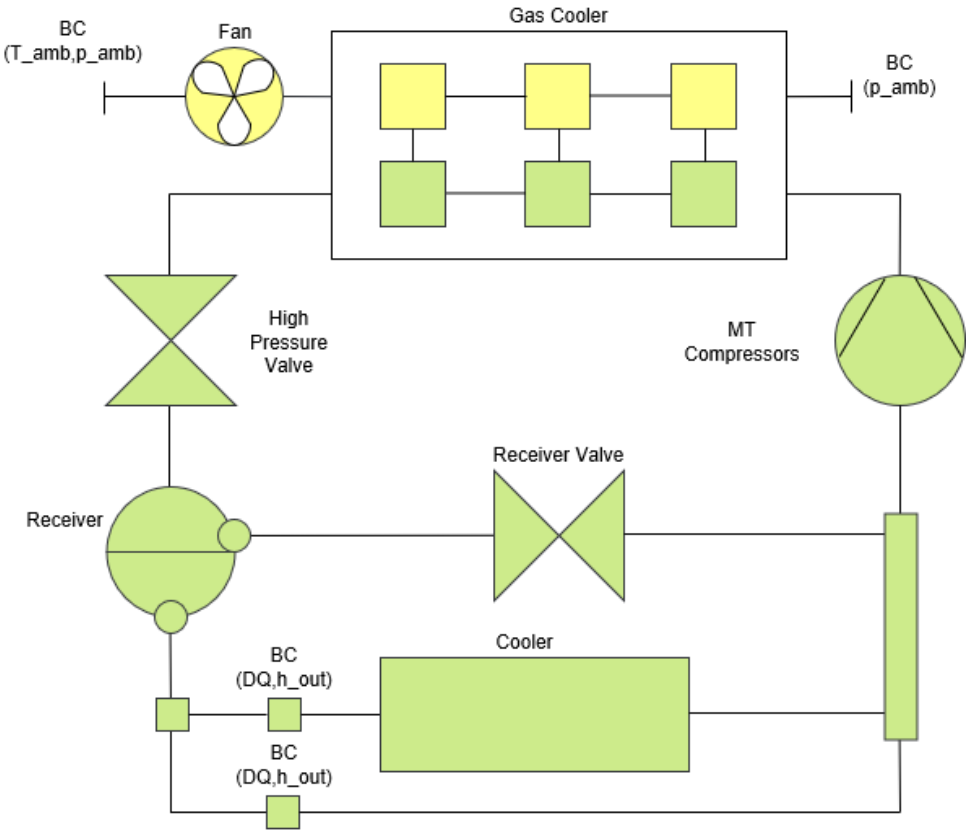

Figure 4.1: Two Level Refrigeration System

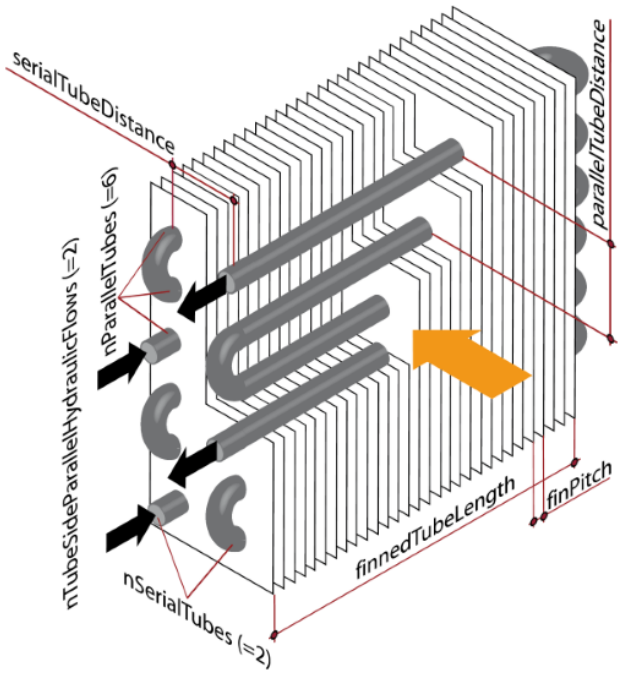

Figure 4.2: Physical structure of a gas cooler

The system can be described in the general form

$$\dot{x} = f(x, \mathrm{BC}(t), u(t)), \tag{4.1}$$

where x is the state vector, BC(t) are the boundary conditions, and $u(t)$ are the input signals (control output), being zero order holded. These can be described as

$$x = \begin{bmatrix} \dot{V}_A \\ p_i \\ h_i \\ \rho_i \\ \dot{m}_{i+1,i} \\ T_{A,i} \\ \dot{m}_{HP} \\ p_R \\ h_R \\ \rho_R \\ \dot{m}_{RP} \\ p_C \\ h_C \\ \rho_C \\ \dot{m}_{MT} \end{bmatrix}, \qquad u(t) = \begin{bmatrix} CR_A(t) \\ CR_{HP}(t) \\ CR_{RP}(t) \\ f_{MT}(t) \end{bmatrix}, \qquad BC(t) = \begin{bmatrix} T_{A,0}(t) \\ p_0(t) \\ \dot{Q}_C(t) \\ h_{C,in}(t) \\ \dot{Q}_{freezer}(t) \\ h_{F,out}(t) \end{bmatrix}, \tag{4.2}$$

where the index A denotes air, index HP high pressure, index R receiver, index RP receiver pressure (receiver valve), index C cooler, index F freezer, index 0 ambient (air), index in inlet, index out outlet, index MT Medium Temperature (denoted for the pressure level of the cooler and the implemented compressors). Index i denotes the i^{th} cell of the gas cooler, having $x_i \quad \epsilon \quad \mathbb{R}^{n_{cell}}, \quad i = 1...n_{cell}$ (parameter n_{cell} being the number of cells), and $\dot{m}_{i+1,i}$ denotes the mass flow from cell i to $i+1$.

The fan is controlling the air volume flow through the gas cooler with the capacity ratio multiplied by the maximum volume flow (this is a significant simplification, since the fan has some hysteresis for low levels of volume flow and saturates above a value):

$$\ddot{V}_A = \frac{1}{\tau_A}\left(-\dot{V}_A + \dot{V}_{A,max}\mathrm{CR}_A\right). \tag{4.3}$$

The volumes in the system can be described by the first three equations from Section 3, with no technical work. The mass balance equation shows that the density is the "least dependent" state,

$$\dot{\rho}_i = \frac{1}{V_i}\left(\sum_j \dot{m}_{in,i,j} - \sum_k \dot{m}_{out,i,k},\right) \tag{4.4}$$

while the change of the pressure and the enthalpy are joint. They depend on not only each other (being described by the joint matrix J_i), but on the change in density, and the excitation $\dot{\Psi}$ as well (see the equations below too). Note that the element $-\frac{p_i}{\rho_i}\dot{\rho}_i$ is a dominating element (resulting in dynamics defining the speed of sound), but is filtered out by the low pass filters in the system. Not applying the low pass filters stiffened the system too much for the solvers.

$$\begin{aligned} J_i \begin{bmatrix} \dot{p}_i \\ \dot{h}_i \end{bmatrix} &= \begin{bmatrix} \frac{\dot{\Psi}_i}{V_i} - \frac{p_i}{\rho_i}\dot{\rho}_i \\ \dot{\rho}_i \end{bmatrix} \\ J_i = \begin{bmatrix} -1 & \rho_i \\ \frac{\partial\rho}{\partial p}|_{h\,i} & \frac{\partial\rho}{\partial h}|_{p\,i} \end{bmatrix} \qquad & \dot{\Psi}_i = \sum_j \dot{m}_{in,i,j}h_{in,j} - \sum_k \dot{m}_{out,i,k}h_{out,k} + \dot{Q}_i \end{aligned} \tag{4.5}$$

The change in the mass flow rate between cells is induced by the pressure difference between

cells, assuming constant hydraulic resistance $R_{hyd,cell}$:

$$\ddot{m}_{i+1,i} = \frac{1}{\tau_R}\left(-\dot{m}_{i+1,i} + \frac{1}{\Delta R_{hyd}}\sqrt{\rho_i(p_i - p_{i+1})}\right). \tag{4.6}$$

The temperature of the air cell is a weighted average of the inlet air temperature and the refrigerant temperature, where the weight is the ratio of the conductances of the moving air and the convection between the fins of the gas cooler and the air,

$$\begin{aligned} \dot{T}_{A,i} &= \frac{1}{\tau_T}\left(-T_{A,i} + \frac{1}{w+1}T_i + \frac{w}{w+1}T_{A,i-1}\right), \\ w &= \frac{\rho_A \dot{V}_A c_p}{\Delta\sigma_{conv}}. \end{aligned} \tag{4.7}$$

The standard air density is denoted by $\rho_A = 1.2$ kg m^{-3}, and the isobaric specific heat capacity at our temperature range is $c_p = 1000$ J (kg K)$^{-1}$. The index *conv* refers to the convection between the fins and the air.

The valves are generally described by a maximum opening K_v value, and their mass flow rate is induced by the provided capacity ratio (actuation) and the pressure difference on the two sides,

$$\ddot{m}_V = \frac{1}{\tau_V}\left(-\dot{m}_V + K_v\sqrt{\rho_{in}(p_{in} - p_{out})}\mathrm{CR}_V\right). \tag{4.8}$$

The compressor has a small cabin, in which it compresses the air from the suction to the discharge side. The effective volume of the cabin is called the displacement V_{MT}, and the compressor appears to be a current generator for the system (being constrained by density changed on the suction side), being controlled by frequency f_{MT}. (Note that the necessary torque is not considered in the analysis, but it could be calculated from the technical work in the static version of the energy balance equation, assuming $\dot{Q} = 0$.) The descriptive equation is below.

$$\ddot{m}_{MT} = \frac{1}{\tau_{MT}}\left(-\dot{m}_{MT} + \rho_{MT}V_{MT}f_{MT}\right). \tag{4.9}$$

Chapter 5

Intermediate calculations in the simulation

The heat transfer to a cell, the thermal conductance and the convection between the fins and the air volume are calculated by the equations below. Note that the thermal conductance is assumed to be the same all along the tube, and that the convection has a minimum value σ_0, which is the natural convection, and is then increased by the volume flow rate multiplied by the constant k_{conv}. The heat transfer can be imagined as two serially connected resistances, which both depend on the

volume flow rate, see Figure 5.1.

$$\begin{aligned} \dot{Q}_i &= (T_{A,i} - T_i)\,\Delta\sigma_{conv} \\ \Delta\sigma_{conv} &= \frac{\sigma_{conv}}{n_{cell}}, \\ \sigma_{conv} &= \sigma_{conv,0} + k_{conv}\dot{V}_A. \end{aligned} \tag{5.1}$$

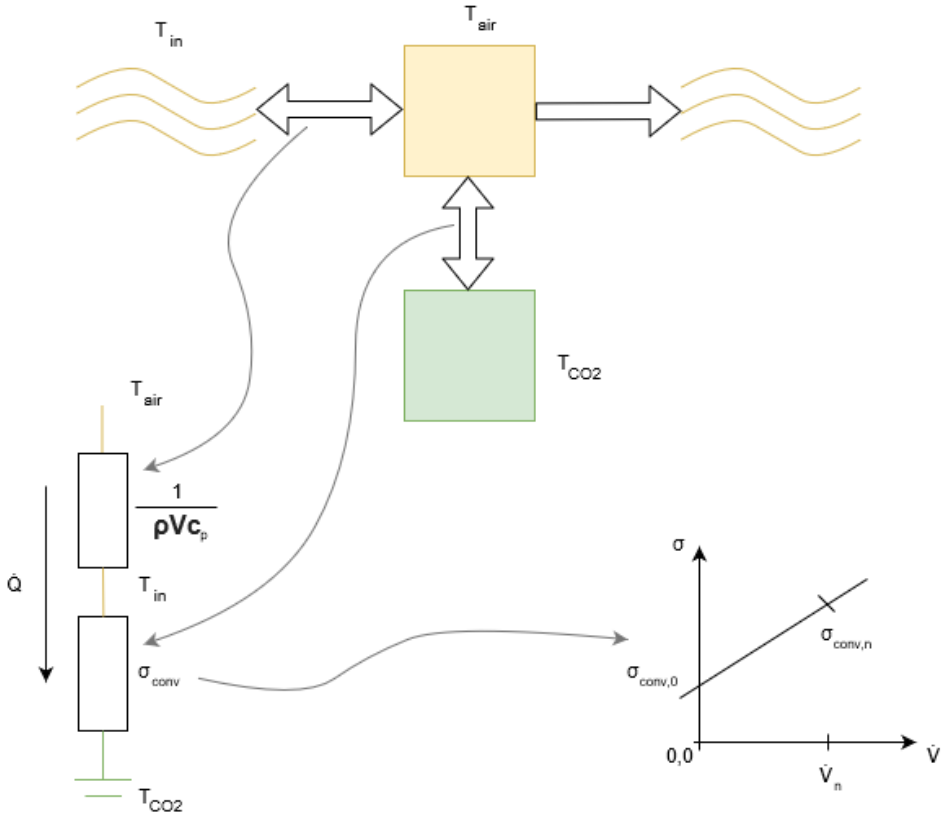

Figure 5.1: Modelling of heat transfer

The receiver is insulated, so $\dot{Q} = 0$ in the energy balance equation. There are three mass flows, the inlet, and the gas and the liquid outlets. The outlet enthalpies are read from the ph diagram, assuming equilibrium -which is a close assumption as the receiver can be considered as a buffer-, as well as the outlet densities.

The receiver joint fulfills the mass balance equation with no accumulation. The subscripts are liquid L, freezer F, and C, in for the inlet side of the cooler evaporator (but after the boundary condition).

$$\dot{m}_L = \dot{m}_F + \dot{m}_{C,in} \tag{5.2}$$

The boundary conditions assume constant heat flow rate into the system (people are opening the door of the cooler and the freezer cabinets constantly, and the changing temperature difference is neglected here), and that the temperature (or superheat) control on the outlet side of the evaporators are perfect - the outlet enthalpies are the reference values, having the correct pressure. The

subscript for the outlet of the freezer is F, out.

$$\begin{aligned} \dot{m}_F &= \frac{\dot{Q}_F}{h_{F,out} - h_L} \\ \dot{m}_C &= \frac{\dot{Q}_C}{h_{C,in} - h_L} \end{aligned} \tag{5.3}$$

The MT joint fulfills the mass balance and energy balance equations with no accumulation. The new subscripts are notations for suction side of the compressor MT, in and outlet of the cooler volume C, out.

$$\begin{aligned} \dot{m}_{C,out} &= \dot{m}_{MT} - \dot{m}_{RP} - \dot{m}_F \\ h_{MT,in} &= \frac{\dot{m}_{RP} h_{RP} + \dot{m}_{C,out} h_c + \dot{m}_F h_{F,out}}{\dot{m}_{MT}} \end{aligned} \tag{5.4}$$

For the valves, it can be deduced from the static version of the energy balance equation (assuming no technical work and heat flow rate), that the outlet enthalpy is the same as the inlet enthalpy.

$$h_V = h_{in} \tag{5.5}$$

An ideal compressor raises pressure by staying at the same entropy, then the ph diagram could be used to find the outlet enthalpy. However, the isentropic efficiency $\eta_s \approx 0.6$ is to be considered as a value from the industry (for a particular *Bitzer* compressor).

$$h_{MT} = h_{MT,in} + \frac{h_{MT,id}(h_{MT,in}, s_{in}) - h_{MT,in}}{\eta_s} \tag{5.6}$$

Chapter 6

Deduction of equations, and initial calculations

In order to find an expression of the time derivative of the energy in terms of our states, we use the definition of enthalpy $h = u + pv$ and the chain rule when differentiating $v = \frac{1}{\rho}$.

$$\begin{aligned} \rho V \dot{u} &= \dot{\Psi} \\ \dot{u} &= \frac{\partial(h - pv)}{\partial t} = \dot{h} - \frac{1}{\rho}\dot{p} + \frac{p}{\rho^2}\dot{\rho} \end{aligned} \tag{6.1}$$

The partial derivatives in Equation 4.5 are derived using the chain rule of partial differentiation, hence

$$\dot{\rho} = \frac{\partial \rho}{\partial p}|_h \dot{p} + \frac{\partial \rho}{\partial h}|_p \dot{h}. \tag{6.2}$$

To deduce the interaction between the refrigerant and the air cells, four stages of thermal resistance shall be considered. Approaching from the refrigerant to the gas inlet of a cell, these are the following.

- The convection between the refrigerant and the copper tube wall.
- The resistance of the copper tube and the aluminium fins.
- The convection between the aluminium fins and the gas (air).
- The heat resistance of air, described by the (static) energy balance equation of air.

In practice, the first two are negligible compared to the last two, there are more orders of magnitude differences in thermal resistance. This is why

$$\frac{1}{\sigma} = \sum_{j=1}^{4} \frac{1}{\sigma_j} \approx \frac{1}{\sigma_{conv}} + \frac{1}{\rho_A \dot{V}_A c_p}. \tag{6.3}$$

For the two convection parameters, we use the the two equations below. The first one uses the arbitrary design parameter $\alpha = 5$ (which is taken as a parameter used to produce reasonable simulations in the Modelica model), while the second equation approximates the nominal convection from the nominal heat flow rate $\dot{Q} = 74.3kW$, and the pitch temperature difference (this is a value that so called controls, saturates the maximal heat transfer). Not having more information about the distribution, the latter is assumed to be at the refrigerant outlet, begin $\Delta T_{pitch} = 3\ K$. Note that $\Delta T_{log} = \frac{\Delta T_{out} - \Delta T_{in}}{\log \frac{\Delta T_{out}}{\Delta T_{in}}}$ would be another way of approximating the temperature difference, in case of exponential curves. However, our curves are not exactly exponential, and according to experiments with the simulation, the value $\Delta T_{log} \approx 20\ K$ gives around 10% lower heat transfer than the required one.

$$\begin{aligned} \sigma_{conv,0} &\approx \frac{\sigma_{conv,n}}{\alpha} \\ \sigma_{conv,n} &= \frac{\dot{Q}_n}{\Delta T_{pitch}} = \sigma_{conv,0} + k_{conv} \dot{V}_{A,n} \end{aligned} \tag{6.4}$$

For discretization, the manner of the heat conduction was considered. Assume that the convection is the product of the area A_{fin} and the heat transfer coefficient U, the latter being only dependent on the materials of fin and air, then having equal heat transfer areas for every cell gives

$$\begin{aligned} \sigma_{conv} &= U A_{fin}, \quad \text{hence} \\ \Delta\sigma_{conv} &= U \Delta A_{fin} = \frac{\sigma_{conv}}{n_{cell}}. \end{aligned} \tag{6.5}$$

Two equations are used to deduce all the results of Equation 5.1, the (static) energy balance equation for air (derived from the definition of the specific heat capacity $c_p = \frac{Q}{\Delta T \dot{m}}$),

$$\dot{Q}_i = (T_{A,i-1} - T_A)\rho_A \dot{V}_A c_p + \frac{\mathrm{d}(T_{A,i-1} - T_A)}{\mathrm{d}t} \rho_A \Delta V c_p, \tag{6.6}$$

and the assumption, that the heat flow rate can be approximated as in electronics, as the product of the temperature difference (intensive variable or potential) and the convection, and assuming some capacitance,

$$\begin{aligned} \dot{Q}_i &= (T_{A,i} - T_i)\sigma_{conv} + \frac{\mathrm{d}(T_{A,i-1} - T_A)}{\mathrm{d}t} \tau_A, \\ \tau_A &\equiv \rho_A \Delta V c_p = \rho_A \frac{A_{GC} H_{GC}}{n_{cell}} c_p, \end{aligned} \tag{6.7}$$

where A_{GC} is the horizontal area and H_{GC} is the height of the gas cooler, n_{cell} is the number of cells chosen to model the gas cooler. For simplicity, the time constant in Equation 6.7 is assumed to be the same as the time constant in Equation 6.6.

The fluid dynamics in the gas cooler are considered how it is described in the chapter *Dimensioning of tubes* in [2]. Even with rough estimations, it turns out that the flow is turbulent. The calculation of the friction factor f of the pipe happened with iterating the definition for Reynold's number (denoted as Re), the Gnielinski correlation, and the Darcy–Weisbach equation, until convergence was achieved. Note that the range of Gnielinski correlation is $2300 < \mathrm{Re} < 5e6$. The iterated equations are

$$\begin{aligned} \mathrm{Re} &= \frac{4\dot{m}}{\pi d\mu}, \\ f &= \frac{0.5}{(0.79 \log \mathrm{Re} - 1.64)^2}, \\ R_{hyd} &= \sqrt{\frac{16fl}{d^5\pi^2}}, \\ \dot{m} &= \frac{1}{R_{hyd}}\sqrt{\rho \Delta p}. \end{aligned} \tag{6.8}$$

where d is the diameter, l is the length, Δp is the pressure difference along the two end points of the tube, and the dynamic viscosity $\mu = \rho\nu$, and the kinematic viscosity ν is the property of the fluid. Then the last equation was used to determine the mass flow rate between cells. The discretization was done by considering that only the length of a tube slice changes by altering the number of cells,

$$\Delta R_{hyd} = \sqrt{\frac{16f\Delta l}{d^5\pi^2}} = \sqrt{\frac{16fl}{d^5\pi^2 n_{cell}}}. \tag{6.9}$$

Chapter 7

Architecture of code

The code was written in Matlab programming language. Object oriented approach was chosen in order to enhance readability. This also gives a better hand in making the code modular, therefore carrying out modifications is quicker.

Since the equations are described by assignments, and the solver is explicit, input-output approach was used in coding (see Figure 7.1). The volume components took mass flow rates and enthalpy inlet as input, and the outputs were the density, the pressure and the enthalpy. The concentrated elements take these states as input, and determine mass flow rates and outlet enthalpies. This can happen with resistance, or current generation.

Output means the inner state of the volume component. This makes the number of cells in discretization an important influencer of the results. Consider the case, when we calculate the temperature in the last cell of the gas cooler, which is dependent on the total pressure and total enthalpy

of the cell. Then the control, which takes the outlet temperature, as the temperature of the last cell, is going to cool the gas cooler too much (the fans run too fast), if the number of cells are too low.

The components are often applied as multiple ones in industry, for better efficiency (for example, for compressors, there is a bigger compressor, which is running continuously on full capacity, and at least one smaller, which actually controls the set point). These features are neglected and the components were concentrated with regard to the thermodynamic point of view.

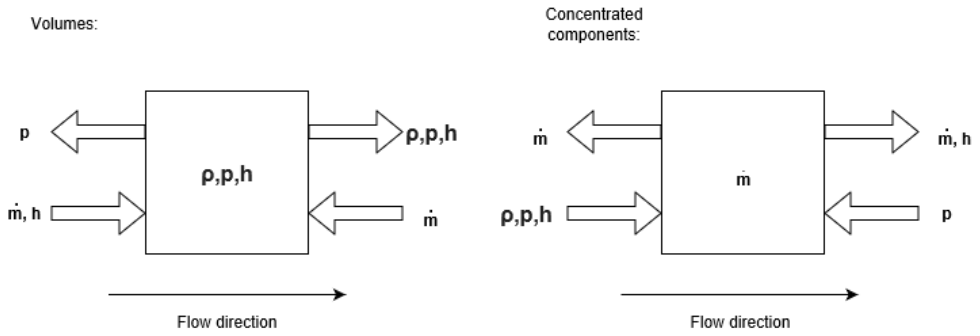

Figure 7.1: The I/O approach of the coding

The architecture can be seen in Figure 7.2. The simulation was parameterized by a synchronised controller sampling time, which resulted in immediate change of actuation. Distributed PI controllers were used. In the iterations of the loop, after calculating the control outputs, the PhysicalPlant object is used to calculate state propagation by the next sampling instant. For that, it composes of a Parts object, which associates to several modelled components. Some of these components are extended from a base component, like Tank, which includes the basic volume equation. It is not shown on the diagram, but in these cases polymorphism is used, meaning that some of the methods of the Tank object were redefined. The PhysicalPlant object also composes of the Connect object, which can be considered as a method, since it only includes a parsing function, which equates certain inlets of a component to certain states (outputs) of another component.

Furthermore, PhysicalPlant composes of TimeStep method as well. This uses the explicit solver ode15s from Matlab. This is a solver for handling stiff differential equations well (we have dynamics of both fluid and thermodynamics, which happen on different time scales). The Process function is provided (aggregated) to this function, which describes the algebraic constraints and the differential equations. The function PostProcess is also aggregated to the TimeStep method, and it manages state update and the creation of records during the simulation.

For state conversion and calculating partial differentials, the CoolProp library http://www.coolprop.org/ (click) was used. It uses the iteration of analytic equations in order to determine the necessary attributes.

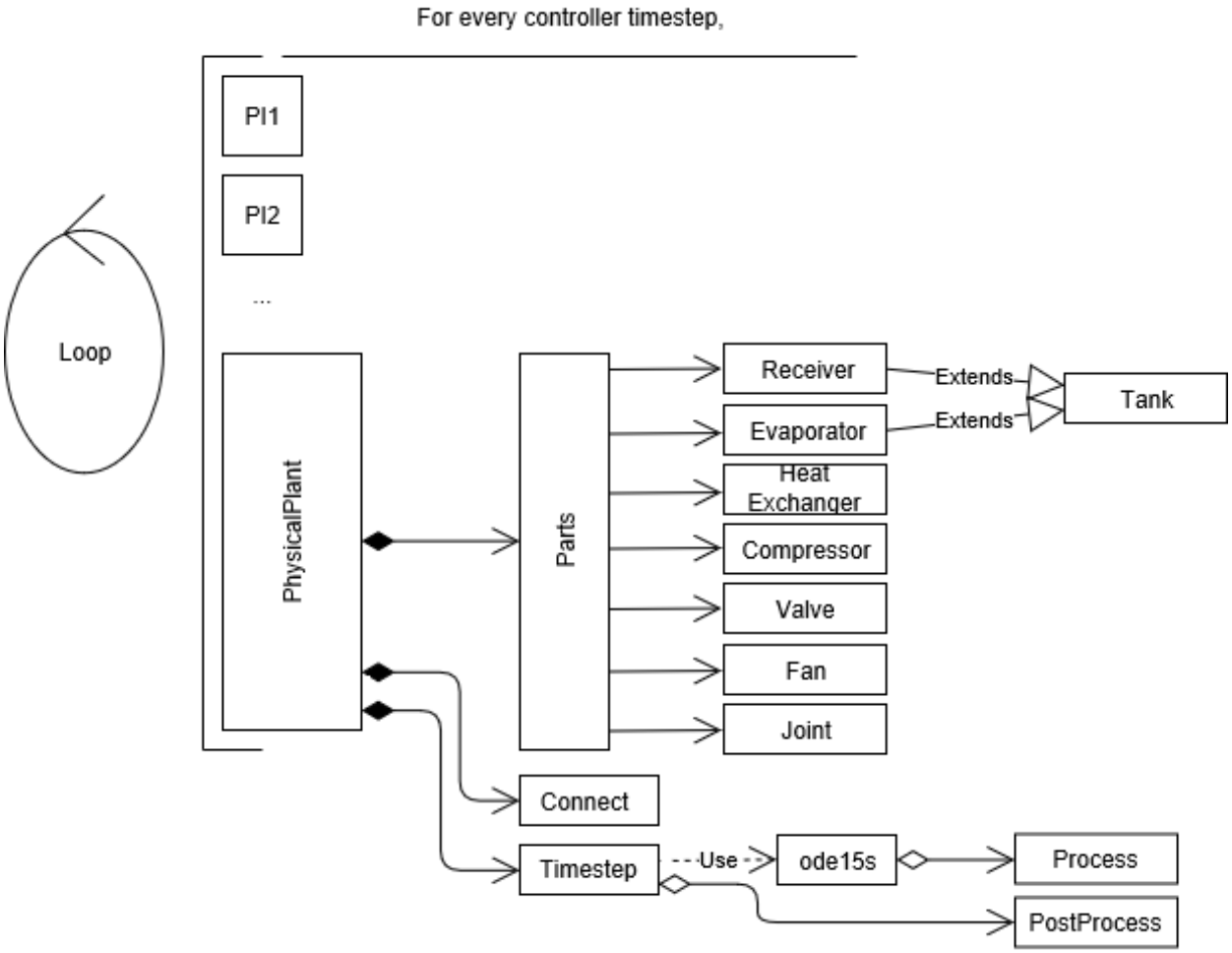

Figure 7.2: Object Oriented Architecture

Chapter 8

Results of the simulations

In this chapter, the results of the simulations are depicted. They show a experimental simulation for the entire system, a stress tested simulation for only the high pressure side, and the stationary discrete temperature distribution along the gas cooler for this latter simulation. Finally, the observed sound dynamics is presented.

Distributed PI controllers (with no anti-windup) were used. The expected stationary values of the simulations are described in Table 8.1. These values correspond to an actual industrial measurement, and served as the design point for the simulation. The simulations are started with the stationary boundary values, however, the pressure and enthalpy distributions in the gas cooler are initialized as linear, which leads to the "reordering" of the energy in the system, resulting in

the initial transients.

Notation	**Description**	**Expected value**	**Reference / Boundary condition**
p_i	Pressure of each gas cooler cells	Last one: 85e5 Pa	reference for HP valve
h_i	Enthalpy of each gas cooler cells	Last one: 298e3 J/kg	-
T_i	Temperature of each gas cooler CO2 cells	Last one: 30 °C	reference for fan
$T_{A,i}$	Temperature of each gas cooler air cells	First one: 48.6 °C	-
ρ_i	Density of each gas cooler cells	-	-
$\dot{m}_{HP}$	Mass flow rate of high pressure valve	0.321 kg/s	-
p_R	Pressure of receiver	38e5 Pa	reference for RP valve
h_R	Enthalpy of receiver	-	-
$\dot{m}_{RP}$	Mass flow rate of receiver pressure valve	0.123 kg/s	-
p_C	Pressure of cooler evaporator	30e5 Pa	reference for MT compressors
h_C	Enthalpy of cooler evaporator	440e3 J/kg	becomes boundary condition in steady state
$\dot{m}_{MT}$	Mass flow rate of MT compressor	0.321 kg/s	-
$\dot{V}_A$	Volume flow rate of fan	3.33 m^3/s	-
$\dot{Q}_{GC}$	Heat flow rate through the gas cooler	74.3e3 W	-

Table 8.1: Table of expected / nominal stationary values of the simulations

First, the simulation was prepared for the entire system. This simulation did not include the low pass filter for the interactions within the gas cooler, which slowed down the solver. - therefore the gas cooler was divided into only 3 cells. Furthermore, the sound dynamics part of Equation 4.5 is neglected to have a stable simulation ($-\frac{p}{\rho}\dot{\rho} = 0$). Note that although it is neglected, in its timescale this is the largest component for the equation. However, even this way, the solver is struggling, as the pressures and enthalpies of the gas cooler can take up extreme values. This is due to the high sensitivity to the changes in mass flow rates and hence in the derivatives of the densities in Equation 4.5, as well as the heat flow rates between the air and refrigerant cells. This is one reason why low pass filters were introduced in the following set-up.

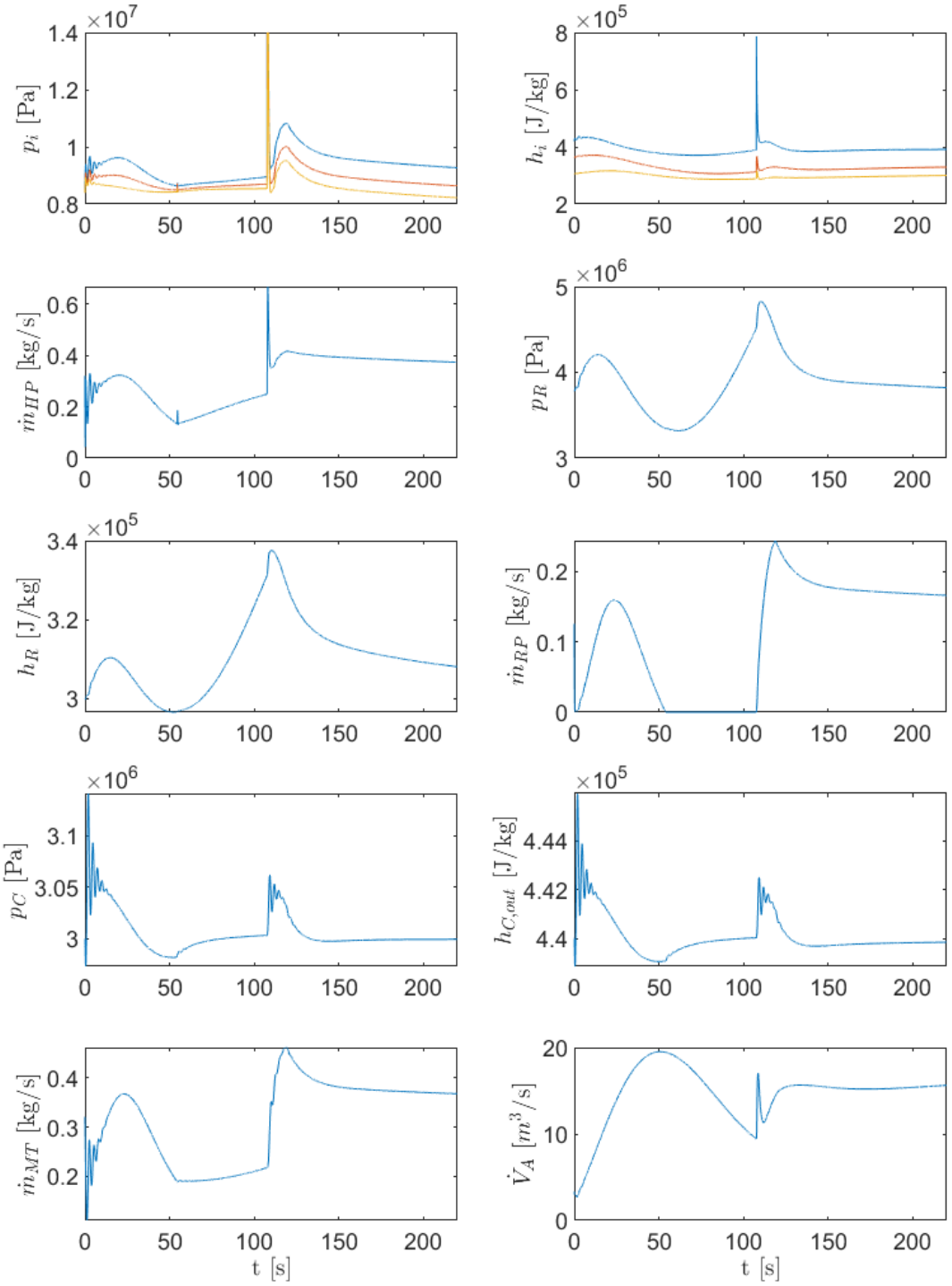

Figure 8.1: Simulation of refrigeration loop ($n_{cell} = 3$), with no low pass filters in the gas cooler

The physical quantities are approaching the expected values. The most important result of the simulation is that the fan needs to be controlled for the system not to unreasonably integrate energy, hence the ambient air temperature cannot be treated as constant. The integration of energy

is caused by the increasing entropic loss for the compressor, if the inlet enthalpy is increased. The integration of energy was generally observed firstly in pressure changes in the receiver (often it filled up with liquid, or became empty). This shows the receiver as a good implementation for having a buffer in the system, as well as a good source of information, if the control does not work properly. Note that due to the wrong setting of the convection, much larger volume flow rate was needed, than expected. This is due to that the logarithmic temperature difference was used in Equation 6.4, instead of the pitch temperature difference.

After that a better resolved version of the gas cooler was made, taking low pass filters and sound dynamics into consideration (most of the latter was filtered out after some seconds of oscillations). The simulation speed significantly increased, making it possible to test it with 10 cells. This resulted in stationary results, which are very close to the nominal values.

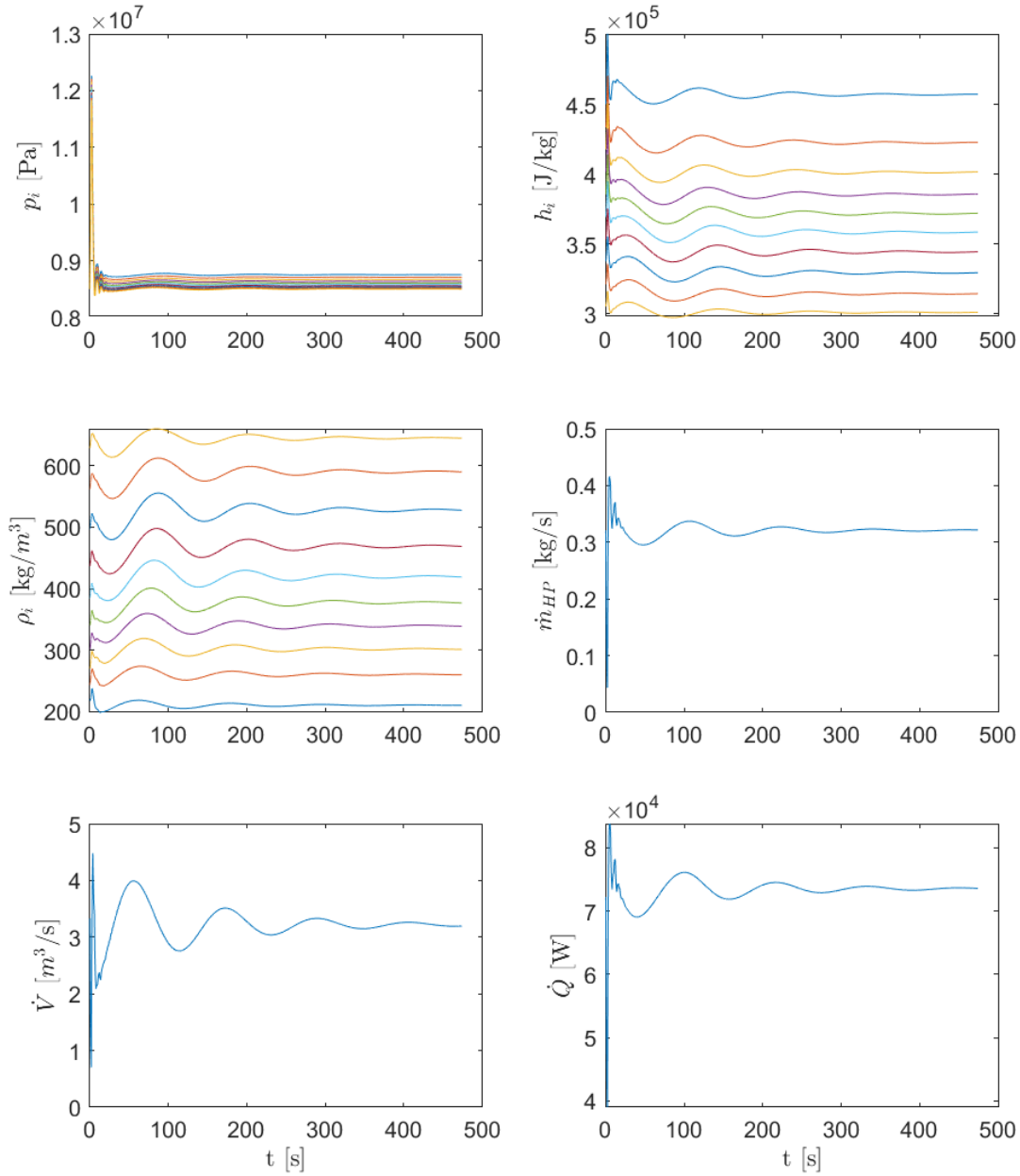

Figure 8.2: Simulation of the gas cooler ($n_{cell} = 10$), high pressure valve and fan

The stationary temperature distributions seem correct (see Figure 8.3), reminding of exponential curves, giving approximately 3 degrees as pitch temperature difference. Note, however, that the assumption of expecting this value was used to set-up this simulation. Nevertheless, the reliability of the simulation can be tested by seeing the heat flow rate or volume flow rate values in Figure 8.2, or calculating them from Equation 6.6, and the results are close (approximately 5% error).

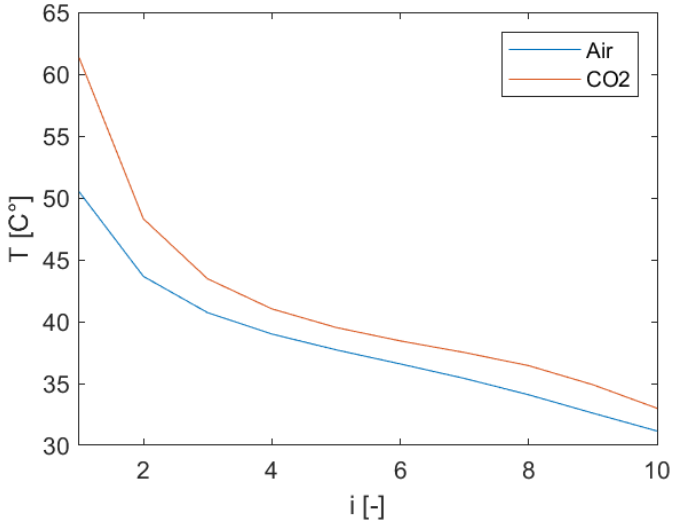

Figure 8.3: Temperature distribution along the cells of the gas cooler

Figure 8.4 depicts the sound dynamics. We can see that the outlet pressures act stiffer than the inlet (higher) ones. After a couple of seconds, these dynamics are completely filtered out, and the hydraulic resistances dominate the pressure differences.

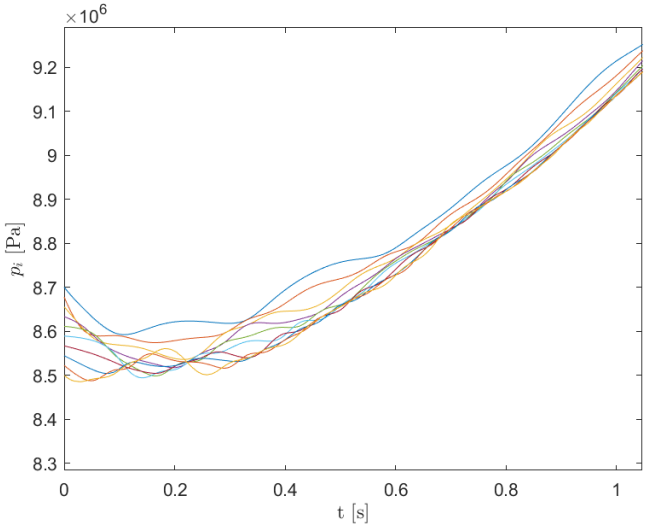

Figure 8.4: Sound dynamics

Chapter 9

Comparison to TIL Suite library

Modelica, Dymola and TIL Suite

Modelica is an object oriented language, which aims at the robust modelling of physical systems. It encompasses mathematical formulations as general equations, and not like assignments, furthermore, it checks for structural singularity and units. It provides the necessary assignments and constraints for a solver.

Dymola is an IDE (integrated development environment) for Modelica. It provides a GUI (graphical unit interface), which avoids a lot of laborious coding. Furthermore, it provides the DASSL solver to solve the differential equations. This is a DAE (differential algebraic equation) solver, which uses an implicit solver on the top of an explicit solver, while fulfilling the necessary algebraic constraints.

TIL Suite is a library for modelling thermodynamic cycles, for example a refrigeration system. It has several components, which are programmed to give a high level of modularity, while they can be partially redeclared to a good extent. A model was developed for an industrial project, using this library, and the results passed a wide range of stress tests, which gives a good basis of comparison for this project.

Comparison of the developed modelling environment and TIL Suite

There are more notable differences in modelling certain elements in TIL Suite and this project. First of all, TIL Suite calculates pressure for a pressure level taking the whole volume of the pressure level into consideration, while it calculates enthalpies and densities locally. Furthermore, the mass flow rates are set as equal to the inlet, and then they are optimized so that the thermodynamic equations are fulfilled as much as possible. This and missing the sound dynamics term $-\frac{p}{\rho}\dot{\rho} = 0$ in Equation 4.5 make TIL Suite entirely neglect fluid dynamics in the gas cooler. TIL Suite also considers only forced convection (the element $\sigma_{conv,0} = 0$ in Equation 6.4). However, it uses much more advanced models to calculate heat resistance, although - according to experiments - these features do not make a big difference for the investigated system. The controllers used are continuous, allowing the simulation to speed up close to steady state.

Different solvers are used; DASSL in Dymola and ODE15S in Matlab. The model set up in TIL Suite is much more sensitive to the initial values (due to using an implicit solver paired with nonlinear equations), and normally provides values far away from the steady state, then the controllers need to force it to the steady state. However, getting closer to that, the simulation becomes significantly faster. The Matlab model provides reliable initial transients, however, due to using explicit solver - which is struggling due to coping with time lag - and the zero-order-hold controllers, it does

not speed up so much close to steady state. It is also notable that the TIL Suite simulation speed does not deteriorate in case of increasing complexity so much, as for the Matlab coded programme.

Both libraries use object oriented architecture, and a high level of modularity. However, since no parsing functions are written to describe the Process and PostProcess function, the Matlab programme is much more laborious to change and is more error prone. Furthermore, the GUI of Dymola makes the modifications more quick, and robust to user errors.

Both libraries use CoolProp library for state conversions and calculating partial differentials, however, it is low level interfaced for TIL Suite and high level interfaced for the Matlab programme, making the former significantly faster (the CPU time of calculating certain partial differentials in high level CoolProp can be 0.05 seconds).

TIL Suite has extreme value handling, for example it limits the enthalpies. It uses a third order smoothing function around zero for the square root function. This is not considered in this project. Furthermore, Modelica checks for units, and for structural singularity, meanwhile in the Matlab programme, assignments give the chance of ignoring structural singularity.

Alternating the models can be challenging in TIL Suite, since almost all equations are protected, and the users are only supposed to change parametric values. This might pop up as a problem for a developer.

Chapter 10

Conclusion and perspectives

Conclusion

In this paper, an entire, reliable model has been set up in Matlab for testing control methods for supermarket refrigeration systems. It provides better initial transients than an industrial alternative, and it models a larger scale of dynamics. However, the slowness of the simulations, the differences in time scales, and the higher potential to commit mistakes, when changing the code, gives a better trust to the TIL Suite library.

There were more achievements of the simulations. The different time scales in a thermodynamic system were simulated with a high fidelity, showing that although sound dynamics can be neglected, hydraulic resistance can actually speed up or slow down heat transfer. It turned out that modelling of the refrigerant loop cannot be done without controlled fans, since constant air temperature on the ambient air side lets the system unreasonably integrate energy (first, in the receiver). It also shows that the receiver is a good source of fault detection, being the buffer of the system.

In summary, the created model acts as a whole (it does not integrate energy, has reasonable time constants, and converges to the correct stationary values). Therefore model based state estimation or control design may be built upon it.

Perspectives

Modelling supermarket refrigeration systems is not done yet. There are missing popular elements, for example the heat recovery unit to heat up the supermarket, the air conditioning unit, a bypass valve of the gas cooler for cold days, in order to avoid the system cool too much, and the IT compressor, which saves on efficiency. Ejectors, as being the more efficient alternatives to high pressure valves, are also not considered, although they are not spread on the market yet.

The most important perspective is that different models are needed to answer different questions. This is why a good, general and robust library provides more use in the long term, then something specific and narrow. For example, an industrial problem of today, is to detect gas loop and recover from it efficiently. Gas loop is caused by an offset or hysteresis in the temperature measurement and results in circulating the gas in vain, while the evaporators are heating up. Finding a correct model for the diagnosis and the energy efficient way of avoiding (or handling) gas loop is an industrial motivation.

Since the slowness is partly caused by the high level usage of the CoolProp library, either low level usage, or parameter estimation may be considered, to achieve higher simulation speed. Furthermore, speed might be increased by implementing the implicit Matlab solver, although it is not as fast as the one provided by Dymola.

Chapter 11

Appendix

There were two ideas discussed during the project, but having no effect on the final results.

- Initially, the I/O approach was not considered, and the simulation was to be solved entirely with an optimization (therefore it would have been a semi-implicit way of solving it). For that FEM (finite element method) could have been used to find the derivatives within the gas cooler. However, boundary conditions are necessary in order to set up the necessary stiffness matrix, which are unknown in the general case.
- A simplified version of deducing the partial derivatives can be used. The analysis is the same that provides the ones in CoolProp library, however, the iterations and a lot of refining factors were neglected. For deducing the results, Van der Waals' equation [1], the definition of entropy and enthalpy, and the Maxwell relations were used. The results were significantly different (sometimes 100%) from the ones provided by the library, therefore the branch of the project was not continued.

Bibliography

[1] Yunus A Cengel and Michael A Boles. "Thermodynamics: an engineering approach". In: *Sea* 1000 (2002), p. 8862.

[2] Eric Granryd. *Refrigerating engineering*. Royal Institute of Technology-Department of Energy Technology, 2005.

[3] T Lajos. "Áramlástan alapjai". In: *Budapest: Műegyetemi kiadó* (2009).

[4] James Rumbaugh et al. *Object-oriented modeling and design*. Vol. 199. 1. Prentice-hall Englewood Cliffs, NJ, 1991.