

Real-Time Simulation of Vapour Compression Cycles

Christian Schulze
TLK-Thermo GmbH,
Hans-Sommer-Str. 5, 38106 Braunschweig, Germany
C.Schulze@TLK-Thermo.de

Manuel Gräber
Technische Universität Braunschweig, Institut für Thermodynamik
Hans-Sommer-Str. 5, 38106 Braunschweig, Germany

Michaela Huhn
Technische Universität Clausthal, Institut für Informatik
Julius-Albert-Str. 4, 38678 Clausthal-Zellerfeld, Germany

Uwe Grätz
ITI GmbH
Webergasse 1, 01067 Dresden, Germany

Abstract

This paper shows a tool chain of a set of ready-to-use tools and libraries that enables the dynamic real-time simulation of vapour compression cycles. A new approach for calculation of fluid properties and numeric efficient component models are applied. As an Hardware in the Loop application a vapour compression cycle is exported to Scale-RT [5] using SimulationX [11] and connected to a hardware PI-Controller in order to realize a superheating control.

Keywords: Real-Time Simulation, Vapour Compression Cycle, Tool Chain

1 Introduction

Whereas so far simulation aimed for conceptual validation in the early concept phase, nowadays we find an increasing need for real-time simulation or even real-time execution of models on micro-controllers. For example Hardware in the Loop (HiL) is an important technique for testing hardware controllers in a simulated environment. It enables the evaluation of a controller on very unlikely test cases which can be implemented easily using simulation, which applies also for vapour compression cycles.

Model-based controllers for a vapour compression cycle can be developed on the basis of a Modelica model. Take the following approaches: A simulation executed on those controllers may on the one hand be used to replace some sensor signals by simulation results; on the other hand the failure of a sensor may be detected by comparing the sensor signals with the simulation results. Furthermore nonlinear model predictive control can be developed based on those Modelica models.

The numeric efficiency of the models and the fluid property calculation methods limits the complexity of the whole model due to the limited calculation speed of the CPU. In case accurate system modelling is needed the numeric efficiency of the component models and the fluid property calculation methods should be improved.

For the application on vapour compression cycles until now there is no numeric efficient ready-to-use tool chain presented that enables efficient simulation of those systems under varying circumstances. In this paper we present a part of the model library and the tool chain developed by the 4 authors of this paper.

The presented tool chain bases on a model library of thermal components, a fluid property library, various simulators and hardware environments as well as a profiling method.

This tool chain and the libraries are exemplified

on a R-407C heat pump cycle with a hardware PI-controller as superheating control, using SimulationX as a compiler and Scale-RT as hardware.

2 Real-Time Fluid Property Library

Profiling results of various thermodynamic systems show that the calculation of fluid properties has a major impact on the model runtime, so the calculation methods for fluid properties must be reconsidered. These results have been gained from the profiling method of the presented tool chain [20]. This method is briefly described in section 4.

For modelling of thermodynamic systems it is usually necessary to have access to the properties of the used media. There is a large number of methods to calculate fluid properties starting from the perfect gas theory to the fundamental equations of state. Those methods differ in the underlying theory, calculation speed, precision, amount of data needed and internal consistency.

For many dynamic simulations of vapour compression cycles a very high precision of the fluid properties is needed to describe the correct pressure levels as well as the correct temperature curve over the heat exchanger length, especially when modelling a superheating control. E.g. the enthalpy of evaporation and the vapour pressure curve have a high influence on those results but they are drawn from the whole fluids property description. Due to this reason fundamental equations of state are employed frequently for simulation of thermodynamic systems like these.

The Helmholtz Potential can be calculated from a fundamental equation of state as a function of density and temperature what from every thermodynamic state variable can be drawn from it. As physical processes often are described by enthalpy differences and pressure differences, the calculated state points frequently are given by pressure and specific enthalpy.

Using a fundamental equation of state the calculation of fluid properties at a given pressure and specific enthalpy requires a numerical solving process. As a result of this, simulations based on fundamental equations of state require a varying CPU work load at a high level and should be avoided for real-time application.

This paper different approach for calculation of fluid properties: instead of Modelica the calculation

methods are implemented in C. This way the the solving methods and solver parameters may be adapted suitable for this application therefore performance is gained. As depicted in figure 7 the fluid property calculation source code is added to the model source code after the export from SimulationX before compiling.

In order to decrease the CPU work load caused by the fluid property calculation other methods instead of fundamental equations of state can be used. In the following we concentrate on calculation methods that are not based on an equation of state.

Fluid properties can be tabulated and interpolated linearly on the basis of equation (1). This way the CPU work load can be reduced significantly but the precision of the results between two tabulated grid points is poor due to the linear approximation. As a countermeasure the distance between two tabulated grid points must be reduced to a minimum so the amount of data is very high. Either the thermodynamic properties or the results and derivatives of a thermodynamic potential [1] can be tabulated.

$$T(p, h) = \sum_{i=0}^1 \sum_{j=0}^1 a_{ij} p^i h^j \quad (1)$$

In addition to the precision of a single value the internal consistency of linearly interpolated data between different values is poor, too. For example if temperature and the specific heat capacity are tabulated then the specific heat capacity calculated from the difference quotient of two temperatures differs from the linear interpolated heat capacity. So $\int c_p dT$ does not approach Δh and hence the energy balance may be violated depending on the model equations.

Spline interpolation is another way to close the gap between to tabulated grid points. Spline interpolation requires the tabulation of the spline function coefficients (see equation (2)). These coefficients can be calculated from the tabulated grid points under the constraint that the results during transition from one set of coefficients to another set of coefficients must be continuously differentiable. This technique is discussed and exemplified for the properties of Water by Kunick [12].

$$T(p, h) = \sum_{i=0}^3 \sum_{j=0}^3 a_{ij} p^i h^j \quad (2)$$

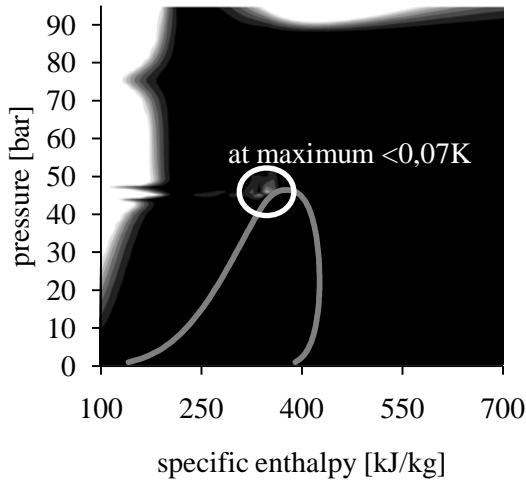


Figure 1: Deviation of temperature compared with Refprop for R-407C PPF.

This paper uses another approach for calculation of one and two phase fluid properties. The basic idea is a functional fit for the enthalpy dependency at various pressure levels in combination with a linear interpolation between those fit functions [21]. This may be interpreted as an array of curves, whereas the gaps between two curves are closed via linear interpolation. Figure 1 shows the error of the temperature calculated from this method compared to the results from Refprop [15] R-407C Pseudo Pure Fluid [14].

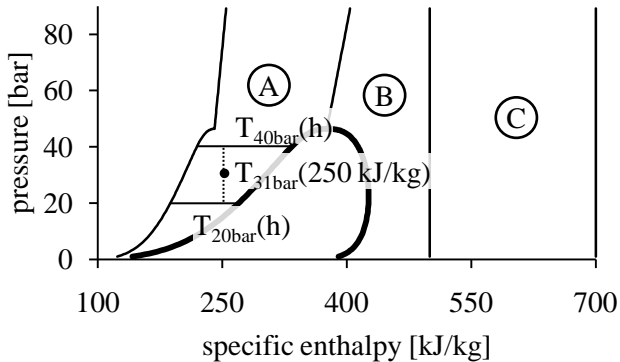


Figure 2: A fluid property is interpolated linearly between the fit equation at upper and lower pressure level, using one set of coefficients for each region A, B and C per pressure level.

Figure 2 illustrates the method of calculating fluid properties using equation (3): The coefficients for the fit equation are tabled for about 200 pressure levels from the triple point up to 85 bars, but the number of pressure levels will be reduced in future. A separate set of coefficients is needed for each region A, B and C. The outstanding fea-

ture of this method compared to the simple linear interpolation is its significantly improved consistency and the reduction of required data.

$$T(p, h) = T_{p_i}(h) + \frac{T_{p_j}(h) - T_{p_i}(h)}{p_j - p_i}(p - p_i) \quad (3)$$

3 Component Library for Real-Time Applications

TLK-Thermo GmbH and the Institut für Thermodynamik of TU Braunschweig develop and maintain the Modelica library TIL [19, 8]. This is a component library for modelling complex thermodynamic systems such as heat-pumps, air conditioning and refrigeration cycles as well as organic rankine cycles, and TIL has been used in various academic and industrial research projects. Combined with the real-time fluid property library described in section 2 a lot of systems modelled with TIL can already be run on real-time hardware. Of course, model complexity has to be adapted to the specific needs. For example complex pressure drop and heat transfer correlations will lead to too large computational effort. In this section we will shortly introduce TIL and give an outlook on further model development for real-time applications.

The distinction of different fluid types is a basic design concept of TIL. E.g. there are three valves, one for incompressible liquids, one for ideal gases and another one for refrigerants. Each model can be adapted to the specific physical behaviour. The fluid properties of in incompressible liquid only depend on the temperature, so these models are simpler compared to those of the refrigerant.

3.1 Heat Exchangers

Figure 3 shows the four different heat exchanger modelling approaches implemented in TIL. Whereas the finite volume and moving boundary approach are designed for dynamic simulation, the NTU-Method and finite differences approach simplify the dynamic behavior of the system.

Heat exchanger models based on finite volumes can give a good picture of the real physical behaviour inside. The more cells are used to describe

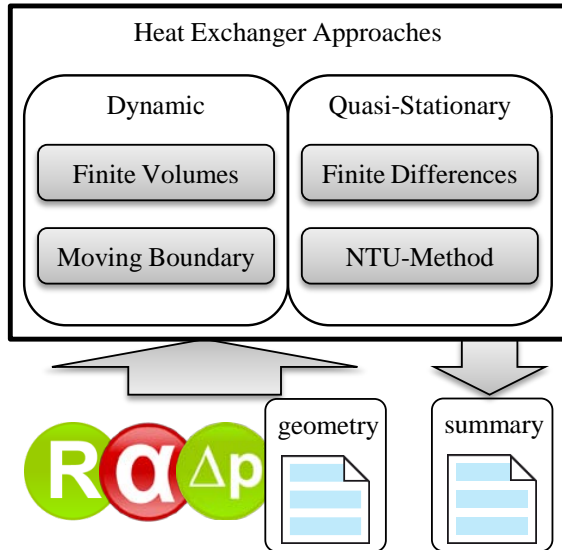


Figure 3: Four Heat Exchanger modelling Approaches implemented in TIL sharing the same interface for fluid properties, pressure drop, heat transfer, geometry and summaries.

the heat exchanger the more detailed will this picture will be. Of course a high number of cells causes a high CPU work load. The general structure of the implemented heat exchanger is shown in figure 4.

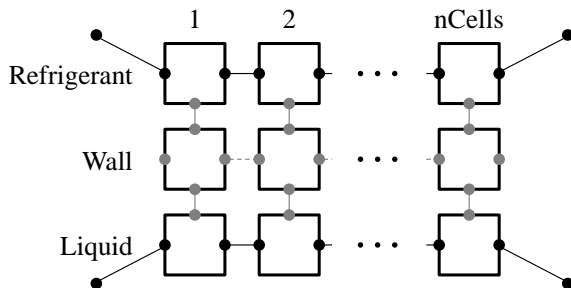


Figure 4: Finite Volume Tube and Tube Heat Exchanger Model based on cell models

In case of an Tube And Tube Heat Exchanger the wall, the refrigerant and the liquid are discretized one dimensionally into the same number of cells as depicted in figure 4. As a simplification the pressure change $\frac{dp}{dt}$ in the cells at the same pressure level is set equal as described by Lemke [13]. As a result of this there is only one continuous time state for the pressure at one pressure level and the CPU work load is reduced significantly.

For dynamic simulation of heat exchangers the moving boundary approach is the second ap-

proach. The Heat Exchanger is lumped into 3 cells, a cell covering the superheated region, one covering the two phase region and another one covering the subcooled region [9]. The lengths of those cells change dynamically, so a suitable heat transfer correlation can be implemented for each section. The precision increases significantly compared to finite volumes with a small number of cells.

The third approach for very fast stationary simulation of heat exchangers is based on the NTU-Method (Number of Transfer Units) [10]. Assuming that the transient effects of heat and mass transfer are negligible the resulting equation system is very small and can be solved quickly. In many cases a stationary model of a vapour compression cycle enables sufficient description of the system behaviour.

The fourth heat exchanger modelling approach is based on the finite difference method. In this approach derivatives of differential equations for temperatures and mass fractions are approximated using finite difference equations. The model is optimized for fast steady state solution at high spatial resolutions and detailed modelling of heat and mass transfer processes [22].

3.2 Efficiency based Compressor

In the compressor model used for the case study of this paper the outlet state is calculated from the number of rotations per second, the displacement, the volumetric and the isentropic efficiency.

$$\dot{V} = \rho_{in} \cdot n \cdot displacement \cdot \eta_{vol} \quad (4)$$

$$h_{out} = h_{in} + \frac{h_{isen} - h_{in}}{\eta_{isen}} \quad (5)$$

3.3 Valve

The Valve is based on Bernoulli's equation. The transition from positive to negative mass flow is smoothed.

$$\dot{m} = \sqrt{2\rho_{in} (p_{in} - p_{out})} \quad (6)$$

3.4 Filling Station

The filling station is a component to control the total mass of refrigerant inside the vapour compression cycle. The mass inside the cycle cannot

be set directly at initialization but this component enables the correction of the mass during simulation.

3.5 Ideal Separator

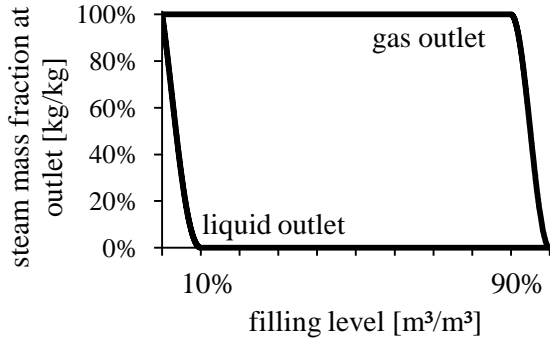


Figure 5: Ideal Separator Characteristic

The ideal separator is a component with a preset volume having 2 outlets and is used to separate liquid and vapour, the characteristic is illustrated in figure 5. In case of a filling level greater than 90% liquid refrigerant will come out of the gas outlet. Between 10% and 90% filling level the separation of liquid and vapour is done perfectly. In case of a filling level below 10% gaseous refrigerant will come out of the liquid outlet. The transition at 10% and 90% filling level is smoothed and hence continuously differentiable.

4 Tool Chain

Many real-time platforms are available to support HiL and RCP such as dSPACE [6], RT-LAB/QNX [17], NI-Veristand [16] or Scale-RT [5]. Scale-RT is a Real Time Operating System based on Linux using the Xenomai Kernel extension running on common desktop PCs. For interaction with the real world, IO-Hardware has to be installed. The RT-Environment used in the case study is a common Desktop PC with a Intel Core2Duo E8400 @3GHz, 4 GB RAM, Scale-RT 5.1.2 and two NI PCIe-6259 as IO-Interface.

Usually Real Time Operating Systems execute a real-time application with a higher priority. The processing of hardware events and the execution of non real-time applications is delayed [7].

Simple IO-Block based controller models can i.e. be created in tools like Matlab Simulink or the

open source project SciCos [3]. Those models can easily be exported to real-time targets. To export Modelica models to a real-time target including external C-Source this external source code has to be added to exported source code before compiling it.

Additional blocks or interfaces have to be implemented into the simulation environment as interfaces to the IO-Hardware. To access the IO-Hardware those blocks have to be instanced and configured in the model.

Depending on the Real Time Operating System (RTOS) tools like Matlab Simulink or SimulationX may either have to offer a hardware driver for each hardware IO-Interface card or can use an internal driver framework of the RTOS. Whereas NI Veristand provides a simulation and hardware framework Scale-RT version 4.x does not provide such a framework. Since version 5.x of Scale-RT a new model framework has been implemented.

SimulationX 3.4 provides access to several IO hardware interfaces for Scale-RT 4.x but unfortunately it did not support the NI PCIe-6259 multi-IO driver for the comedi driver interface [4] yet. This interface has now been added to SimulationX and will soon be available.

On real-time targets usually a fixed step solver is used to integrate the differential algebraic equation system of the whole model. If the calculation time needed exceeds the fixed solver step size an overrun is caused. Depending on the Real Time Operating System this may terminate the simulation. The algebraic loops are solved using a modified Newton solver.

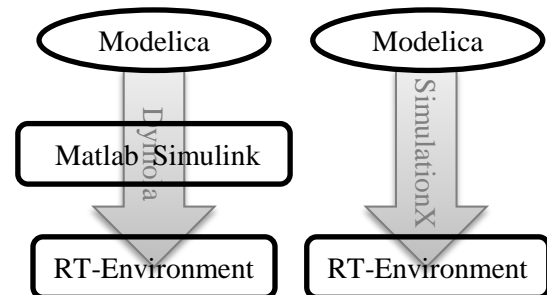


Figure 6: SimulationX enables direct Export to RT-Environment.

In contrast to Dymola SimulationX enables the direct export to a real-time environment in addition to the export via Matlab Simulink / Real Time Workshop as illustrated in Figure 6. The Real Time Workshop is used by many tools as

export interface to real time environments. Details on the export of models from SimulationX to Scale-RT and the optimization performed on it are described in [2].

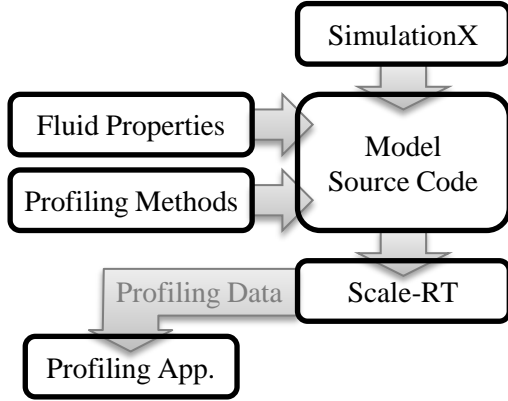


Figure 7: The instrumented model sends profiling data to a secondary application which saves that data

A Profiling method as presented in [20] has been added to this tool chain as well. Profiling of thermodynamic models on the real-time target itself reveals possible weaknesses of the exported model. The process of instrumentation and data saving for the profiling methods is depicted in figure 7. The source code of the exported model has to be modified, on the Real Time Operating System the time-consuming task of saving the data to the hard disk is done by a secondary non real-time application. This profiling method enables a general analysis of the DAE System.

5 Case Study: PI-Controlled R-407C-Cycle

In this section an exemplary application of the above discussed research results is described. We look at a R-407C water to water heat pump cycle. An electric expansion valve and a PI-controller are used to control superheating at evaporator outlet. Real-time simulation experiments are used to tune control parameters.

5.1 Model description

As starting point a model of the vapour compression cycle is built up using the libraries described in sections 2 and 3.

Although finite volumes cause a high CPU work load we decided to use heat exchangers based of 7

finite volumes each, in contrast to Pitchaikani et al. [18] using just one cell. The pressure drop in the heat exchangers is assumed to be negligible.

The model's graphical representation is shown in Figure 8.

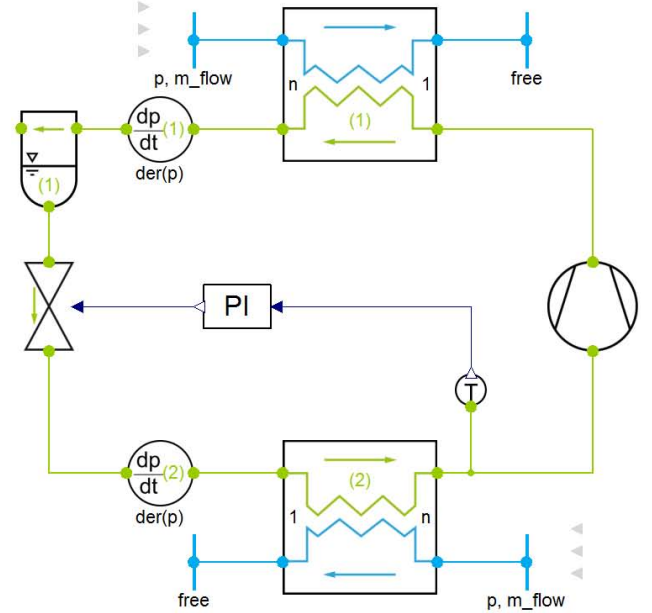


Figure 8: Diagramm of heat pump Modelica model.

Basically the cycle consists of compressor, controlled expansion valve, separator, condenser and evaporator. Superheating is measured at evaporator refrigerant outlet and transmitted to a PI-controller, which computes the expansion valve's opening, according to

$$u = K \left((y - y_s) + \frac{1}{\tau} \int y - y_s dt \right), \quad (7)$$

where u is the relative valve opening, and $y - y_s$ denotes the difference of measured superheating to its set point. Proportional gain K and integral time τ are constant parameters. Suitable values for these parameters are obtained by simulation experiments on a normal PC using the Modelica tool SimulationX. The next step is to use a real hardware controller and connect it to real-time simulated cycle.

5.2 Real-time experiment

The vapour compression cycle model runs on a Scale-RT real-time computer system. A hardware PI-controller is connected via I/O-Boards. Superheating and valve opening are transformed to ana-

integration step size	1ms
sample rate	10ms
number of continuous time states	60
number of integer variables	509
number of real variables	1811
number of external objects (refrigerant properties)	30
sizes of nonlinear systems of equations	2, 3, 9, 11, 3, 1, 9, 11
number of linear systems of equations (size 1)	120
model runtime on RT Environment	$\approx 2.5\text{ms}$

Table 1: Detailed Information about the exported Model

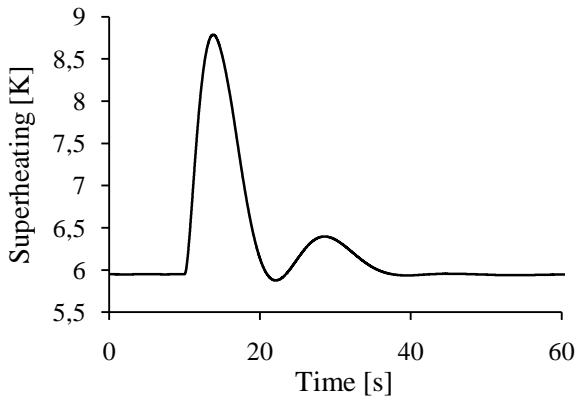


Figure 9: Superheating at evaporator refrigerant outlet.

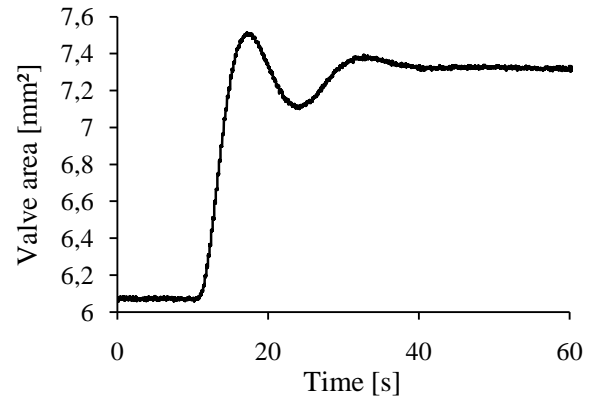


Figure 10: Opening area of expansion valve.

logue voltage signals. Now, the closed-loop performance can be tested under different boundary conditions.

Figure 9 and 10 show an example result. Starting from steady-state condition – superheating is close to its set point of 6K – a step of -5K is applied to the water temperature at evaporator inlet. The systems response is an immediate increase of the superheating and the controller reacts by opening the expansion valve. After one oscillation superheating setpoint is reached again.

6 Conclusion

In this paper a ready-to-use set of libraries as part of a tools chain for real time simulation applications is presented. The TIL Library as well as the TEMO fluid property library are used in this tool chain and exported to Real Time Operating Systems using SimulationX.

As one major contributor to the real time capability of the tool chain a new method for calculation of fluid properties of two-phase refrigerants

is developed and exemplified for R-407C, furthermore other interpolation based calculation methods are described. Four different implemented approaches for heat exchanger modelling – Finite Volumes, Moving Boundary, NTU-Method and Finite Differences – are presented.

Finally as one exemplary case study of the tool chain a numeric efficient and accurate modelling of a vapour compression cycle is presented.

7 Acknowledgement

Most part of this work has been funded by the German Federal Ministry of Education and Research (BMBF) in the project TEMO (grant 01|S08013C).

References

- [1] Trond Andresen. *Mathematical modelica of CO₂ based heat pumping systems*. PhD thesis, Norwegian University of Science and Technology, 2009.

- [2] Torsten Blochwitz and Thomas Beutlich. Real-time simulation of Modelica-based models. In *Proc. 7th Modelica Conference*, pages 386–392. The Modelica Association, 2009.
- [3] Roberto Bucher and Silvano Balemi. Scilab/Scicos and Linux RTAI - A unified approach. In *Proceedings of the IEEE Conference on Control Applications*, Toronto, Canada, August 2005.
- [4] COMEDI. Linux Control and Measurement Device Interface, 2011. URL <http://www.comedi.org>.
- [5] Cosateq GmbH & Co. KG. Scale-RT, 2010. URL <http://www.scale-rt.com/>.
- [6] dSPACE GmbH. dSPACE, 2011. URL <http://www.dSPACE.com/>.
- [7] Philippe Gerum. The XENOMAI Project - Implementing a RTOS emulation framework on GNU/Linux. Technical report, 2002.
- [8] M. Gräber, K. Kosowski, C. Richter, and W. Tegethoff. Modelling of heat pumps with an object-oriented model library for thermodynamic systems. *Mathematical and Computer Modelling of Dynamical Systems*, 16: 195–209, 2010.
- [9] M. Gräber, N. C. Strupp, and W. Tegethoff. Moving Boundary Heat Exchanger Model and Validation Procedure. In *EU-ROSIM Congress on Modelling and Simulation*, Prague, 2010.
- [10] F. P. Incropera, D. P. DeWitt, T. L. Bergman, and A. S. Lavine. *Fundamentals of Heat and Mass Transfer*. John Wiley & Sons US, 6th edition edition, 2006.
- [11] ITI GmbH. SimulationX, 2010. URL <http://www.simulationx.com>.
- [12] Matthias Kunick, Hans-Joachim Kretschmar, and Uwe Gampe. Fast Calculation of Thermodynamic Properties of Water and Steam in Process Modelling using Spline Interpolation. *Proceedings of the 15th International Conference on the Properties of Water and Steam*, 2008.
- [13] Nicholas C. Lemke. *Untersuchung zweistufiger Flüssigkeitskühler mit dem Kältemittel CO₂*. Number 73 in Forschungsberichte des Deutschen Kälte- und Klimatechnischen Vereins. Deutscher Kälte- und Klimatechnischer Verein, Holtzminde, 2005.
- [14] E. W. Lemmon. Pseudo-Pure Fluid Equation of State for the Refrigerant Blends R-410A, R-404A, R-507A, and R-407C. *International Journal of Thermophysics*, Vol. 24, No. 4, 24, 2003.
- [15] M. O. McLinden, S. A. Klein, E. W. Lemmon, and A. P. Peskin. *NIST thermodynamic and transport properties of refrigerants and refrigerant mixtures-REFPROP*. 2008.
- [16] National Instruments. NI VeriStand, 2011. URL <http://www.ni.com/veristand>.
- [17] OPAL-RT. RT-LAB, 2011. URL <http://www.opal-rt.com/product/rt-lab-professional>.
- [18] Anand Pitchaikani, Kingsly Jebakumar S, Shankar Venkataraman, and S. A. Sundaresan. Real-time Drive Cycle Simulation of Automotive Climate Control System. In *Proc. 7th Modelica Conference*, pages 839–846. The Modelica Association, 2009.
- [19] Christoph Richter. *Proposal of New Object-Oriented Equation-Based Model Libraries for Thermodynamic Systems*. PhD thesis, TU Braunschweig, 2008.
- [20] C. Schulze, M. Huhn, and M. Schüler. Profiling of Modelica Real-Time Models. In *Proceedings of the 3rd International Workshop on Equation-Based Object-Oriented Modeling Languages and Tools*, volume 3, pages 23–32, 2010.
- [21] C. Schulze, W. Tegethoff, M. Huhn, and J. Köhler. Numerisch effiziente Berechnungsmethoden für die Stoffeigenschaften von Fluiden für die Systemsimulation. *DKV-Tagungsberichte*, 2010.
- [22] N. C. Strupp, R. M. Kossel, W. Tegethoff, and J. Köhler. Senkung des Kraftstoffverbrauches durch Optimierung der Leerlaufklimatisierung eines PKW mittels Hybridkühlung. In *DKV Tagung*, 2010.