ThermoCycle: A Modelica library for the simulation of thermodynamic systems

Sylvain Quoilin[1], Adriano Desideri[1], Jorrit Wronski[2], Ian Bell[1] and Vincent Lemort[1]

[1] University of Liège, Energy Systems Research Unit
Bâtiment B49, Chemin des Chevreuils 7, 4000 Liège, Belgium
[2] Department of Mechanical Engineering, Technical University of Denmark
Nils Koppels Allé 403, 2800 Kgs. Lyngby, Denmark

Abstract

This paper presents the results of an on-going project to develop ThermoCycle, an open Modelica library for the simulation of low-capacity thermodynamic cycles and thermal systems. Special attention is paid to robustness and simulation speed since dynamic simulations are often limited by numerical constraints and failures, either during initialization or during integration. Furthermore, the use of complex equations of state (EOS) to compute thermodynamic properties significantly decreases the simulation speed. In this paper, the approach adopted in the library to overcome these challenges is presented and discussed.

Keywords: Thermodynamic systems, numerical methods, simulation speed, robustness

1 Introduction

Dynamic simulation of thermodynamic systems is required to evaluate and optimize their response time, or to define, implement and test control strategies. [13, 6, 22]. The Modelica language is well adapted to the formulation of thermo-flow problems, mainly because it is an a-causal language that allows interconnecting the models in a "physical" way [5]. In recent years, several libraries have been developed to model thermodynamic and thermal systems in the Modelica language [6, 27]. A number of libraries are now available to model steam and gas cycles (e.g. ThermoSysPro, Power Plants, Thermal Power, ThermoPower, etc.) or refrigeration systems (TIL, AirConditioning, etc.).

However, not all of them are open-access, and few are able to handle non-conventional working fluids such as refrigerants or fluids used in Organic Rankine Cycle (ORC) systems. Thermophysical substance properties of (moist) air and water are indeed well known and implemented in most simulation tools while those of organic fluids require complex equations of state available only in external libraries such as FluidProp [10], Refprop [16] or CoolProp [1]. Two common solutions when modelling thermodynamic systems requiring external computation of working fluid properties include TIL and ThermoPower:

- TIL is a commercial library for steady-state and transient simulation of thermodynamic systems [26]. The thermodynamic properties are obtained through TILMedia, a library for the calculation of thermophysical substance properties, for example using custom high performance EOS, fast table based bicubic spline interpolation or via an interface for Refprop. It should be noted that TIL-Media is not designed according to the Modelica Media standard but provides an interface with the Modelica.Media library. The TIL library includes a variety of models for thermodynamic components (e.g. heat exchangers, expanders, pumps, etc.).

- The ThermoPower library has proven to be well suited for the modelling of power plants, including ORC systems [8]. It can be coupled to the ExternalMedia library [9], an interface between Modelica and, among others, FluidProp. ThermoPower and ExternalMedia are open-source, FluidProp is freely available but not open-source. The models are mainly designed for large-scale power plants with tube heat exchangers. Smaller systems using e.g. plate heat exchangers are less adapted to the ThermoPower library.

The ThermoCycle Modelica library has been developed for the simulation of thermal plants (heat pumps, steam and gas cycles, etc.) with a focus on smaller-capacity systems. It aims at addressing three typical

challenges inherent to the modelling of thermo-flow systems:

1. Computing the thermophysical substance properties of working fluids

2. Computational efficiency

3. Robustness during initialization and integration

Several numerical methods have been developed and implemented in order to enhance the robustness and the simulation speed of the models during initialization and integration. Furthermore, the computation of the working fluid thermophysical properties is achieved by a strong coupling with the open-source thermodynamic properties database Coolprop [1]. The interface between CoolProp and Modelica is based on the Coolprop2Modelica library, a modified version of the ExternalMedia library [9].

2 The ThermoCycle Modelica library

The ThermoCycle library aims at providing a robust framework to model small-capacity thermodynamic cycles. The goal is to provide an integrated and fully open-source solution ranging from the thermophysical substance properties, using CoolProp, to the simulation of complex systems with their control strategy. In comparison with alternative libraries dedicated to power plants (ThermoPower, ThermoSysPro, Thermal Power), the ThermoCycle library includes various models dedicated to the modelling of smaller-scale thermal systems, such as volumetric compressors models used for the simulation of heat pump or refrigeration cycles. The key features of the Library are the following:

- Designed for system level simulations

- Full compatibility (connector-wise) with libraries such as MST or ThermoPower, use of stream connectors

- Ability to handle reverse flows and flow reversals

- Various numerical robustness strategies implemented in the components and accessible through Boolean parameters

- High readability of the models (limited levels of hierarchical modelling)

The components provided in the library are designed to be as generic as possible. For example, no detailed geometry records should be provided by the user to compute the heat exchanger heat transfer and pressure drop. Instead, the user is expected to build a separate detailed model of the component and extract simplified laws to be used in the Modelica component: for a given fluid and operation conditions range, detailed heat transfer correlations can usually be replaced by simplified equations depending on the flow rate only:

$$ h = h_{nom} \cdot \left(\frac{\dot{M}}{\dot{M}_{nom}} \right)^n \qquad (1) $$

where the n exponent depends on the flow regime (typically 0.6 for laminar and 0.8 for turbulent). This approach aims at increasing the computational efficiency, and also allows using the same model for different types of components. As an example, the same model can readily be used for shell & tubes or plate heat exchangers only be modifying the h_{nom} and n parameters of the simplified heat transfer law.

The same applies for the pressure drops, which can usually be lumped in a single pressure drop on the vapor lines and expressed as a quadratic function of the flow rate. This approach increases the computational efficiency and avoids specific models suitable for one particular component type or geometry only. It is illustrated in Figure 1 for the case of a heat pump: the pressure drops in each line are lumped into two pressure drop components that were calibrated in a separate and more detailed steady-state model. It should be noted that this assumption is valid only if the pressure drops remain limited in both heat exchangers: their influence on the temperature profile should be small compared to the temperature differences and the pinch points of the process.

2.1 Modelling of fluid flows

The finite volume approach is selected as general method to simulate fluid flows. Homogeneous one- and two-phase flow are taken into account, while non-homogeneous, multi-phase flows are not considered, yet. The basic fluid flow component is a cell in which the energy and mass conservation equations are applied (Figure 2). Two types of variables can be distinguished: cell variables and node variables. Node variables are distinguished by the "su" (supply) and "ex" (exhaust) subscripts, and correspond to the inlet and outlet nodes of each cell (Fig. 2). The relation between the cell and node values depends on the discretization scheme. Two schemes are implemented in the cell component, the central difference scheme ($h = (h_{su} + h_{ex})/2$) and the upwind scheme ($h = h_{su}$).
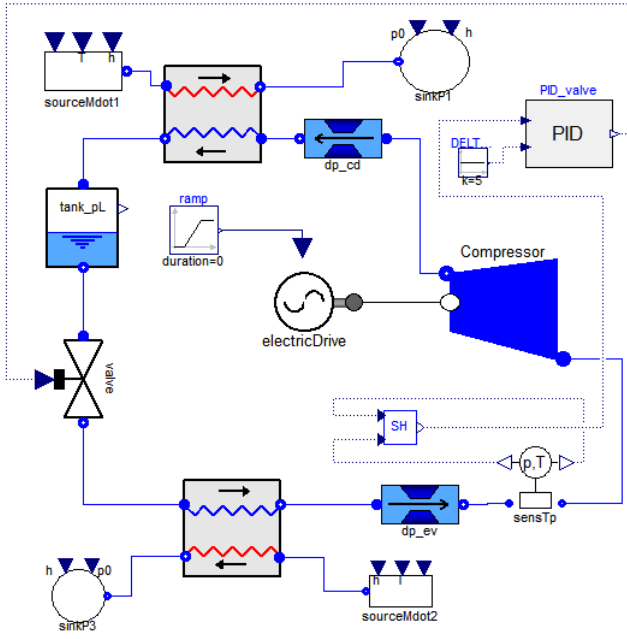
Figure 1: Model of an R407c heat pump

Since the model accounts for flow reversal, a conditional statement is added depending on the flow rates at the inlet and outlet nodes. For the central difference scheme, h_{su} is expressed by Equation 2 (an equivalent equation applies to h_{ex}):

$$h_{su} = \begin{cases} h_{ex}^* & \text{if } \dot{M}_{su} \geq 0 \\ 2 \cdot h - h_{ex} & \text{if } \dot{M}_{su} < 0 \text{ and } \dot{M}_{ex} < 0 \\ h & \text{if } \dot{M}_{su} < 0 \text{ and } \dot{M}_{ex} \geq 0 \end{cases} \quad (2)$$

where the flow rates are defined as positive when the fluid flows in the nominal direction (from "su" to "ex"), and where h_{ex}^* indicates the exhaust node enthalpy of the previous cell.

For the upwind scheme:

$$h_{su} = \begin{cases} h & \text{if } \dot{M}_{su} < 0 \\ h_{ex}^* & \text{if } \dot{M}_{su} \geq 0 \end{cases} \quad (3)$$
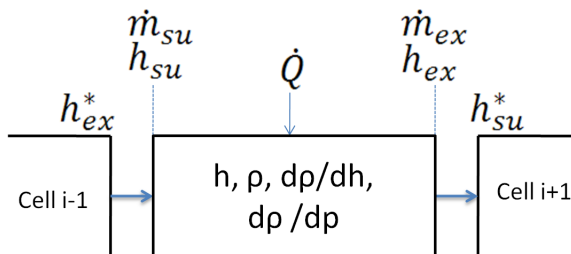

Figure 2: Discretized flow model with cells and node variables

The energy balance for one cell is expressed by [21]:

$$V\rho\frac{dh}{dt} = \dot{M}_{su}(h_{su} - h) - \dot{M}_{ex}(h_{ex} - h) + \dot{Q} + V\frac{dp}{dt} \quad (4)$$

and the mass balance is written as a function of the two differentiated state variable p and h

$$\dot{M}_{ex} - \dot{M}_{su} = V \cdot \frac{d\rho}{dt} = V \cdot \left(\frac{\partial\rho}{\partial h} \cdot \frac{dh}{dt} + \frac{\partial\rho}{\partial p} \cdot \frac{dp}{dt} \right) \quad (5)$$

where $\frac{\partial\rho}{\partial h}$ and $\frac{\partial\rho}{\partial p}$ are considered as thermodynamic properties of the working fluid and are computed in CoolProp. Pressure is considered constant in the cell component.

The cell model has been developed for compressible, incompressible, and constant heat capacity fluid. The overall flow model can be obtained by connecting several cells in series. The final discretization scheme corresponds to a staggered grid, i.e. the thermodynamics states and the state variables (the enthalpies) are calculated inside the cells and the node values ("su" and "ex") are deduced using equations 2 and 3.

2.2 Library structure

The library is organized into different sub packages, including:

1. *Components*, is the central part of the library. It is divided in three sub packages: FluidFlow, Heat-Flow and Units. It contains all the models available in the library from the simple cell model for fluid flow to complete models of heat exchangers, expanders and control units. A more detailed description of this package is presented in sub-section 2.3.

2. *Examples*, contains simulation models where the components of the library are tested. It includes several ORC plant models and it also provides a step by step package where the procedure to build an ORC power unit and an heat pump system is described in detail.

3. *Functions*, includes the empirical correlations used to characterize some of the library models as well as general purpose mathematical functions.

4. *Interfaces*, contains the connectors used for the library components.

5. *Media*, predefines a list of the fluids available in the library.
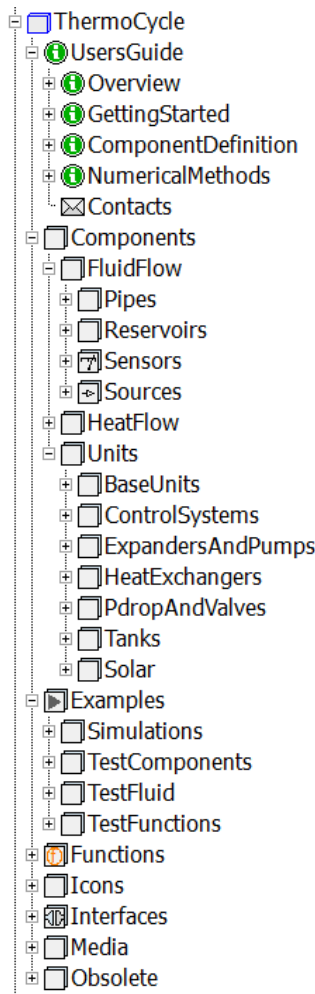
Figure 3 shows an overview of the library structure.

special attention has been paid to their modelling in ThermoCycle. Advanced empirical expressions of the efficiency and filling factor are used and implemented, such as the one proposed by Declaye et al. in the case of an open-drive scroll expander [11].

In the case of volumetric compressors, the efficiencies (isentropic, volumetric) can either be user-defined or based on the standard EN12900, which a well known standard to express the compressor performance as a polynomial function of the evaporation and condensation temperatures.

In addition to these empirical models, more detailed physical models are also available, such as the one proposed by Lemort et al. [17] for scroll machines which takes into account thermal losses, friction losses, internal leakage, internal pressure drop or under and over-expansion (Figure 4). Finally, detailed models of the expansion machine have been included, in which the thermodynamic state of the fluid in the expansion chambers is computed throughout one whole revolution [28]. This type of model is well adapted to the optimization and to the simulation of the component alone, but can hardly be integrated into a more general model because of its significant computational time.
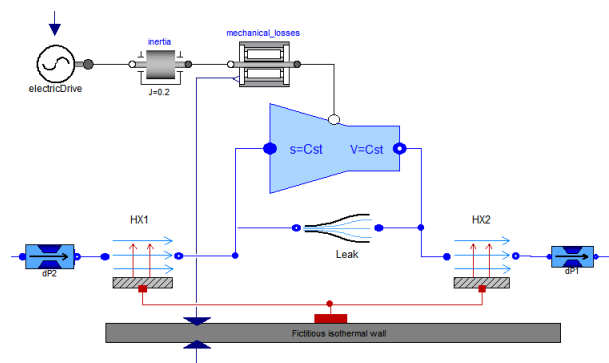
Figure 3: An overview of the library structure from the Dymola graphical user interface

Figure 4: Semi-empirical compressor model

2.3 Physical components

The Components package of the library provides a set of models from the basic cell model to higher-level components commonly used in thermal systems. The most common ones are described hereunder.

2.3.1 Expansion and compression machines

Since the time constants characterizing expansion and compression processes are small compared to that of the heat exchangers [14], the basic pump expander and compressor models are described by empirically derived algebraic correlations. In particular, pumps are modelled by two empirical equations expressing the isentropic and volumetric efficiencies as a function of the operating conditions, e.g. rotational speed and pressure ratio.

Low capacity systems generally use volumetric expansion machines instead of turbomachinery. Therefore,

2.3.2 Heat exchangers

The base component for all heat exchangers is the cell model, see Figure 2. Two types of cells are distinguished: one for compressible (and two-phase) flows, and one for incompressible flows. Heat exchanger models are built by interconnecting these cells in the appropriate manner together with a wall model [27]. Parallel, counter flow and cross flow models for compressible and incompressible fluids are available. More complex structure can also easily be developed by modifying the interconnections between the different fluid flow models.

In addition to the above models based on the finite volumes method, a moving boundaries heat exchanger model for evaporators has also been included. This type of model is known to be more computationally efficient than discretized models and is less likely to be subject to chattering.

2.3.3 Pressure drop and valves

Pressure drops are modelled by lumped parameters used to compute the pressure difference as a function of the flow rate. This function includes a constant term (to model static pressure differences), a linear term (e.g. to model laminar pressure losses) and a quadratic term (to model turbulent pressure losses):

$$\Delta p = \rho \cdot g \cdot h + K \cdot \dot{V} + \frac{1}{A^2} \cdot \frac{\dot{M}^2}{2 \cdot \rho} \qquad (6)$$

where ρ is the fluid density, g is the gravitational constant, $\dot{V}$ is the volume flow rate, $\dot{M}$ is the mass flow rate. The three parameters are h (the static head), K (the linear pressure drop coefficient) and A (the equivalent turbulent orifice area).

Infinite derivative at zero flow in the case of the quadratic term is avoided by using the Modelica *regSquare* function. The fluid is assumed to be incompressible and ambient losses are neglected.

Valves are modelled as a pressure drop whose orifice cross-sectional area can be adjusted by an external signal.

2.3.4 Tanks and liquid receivers

Different tank (fully mixed) and liquid receiver models are available. They are modelled using the energy and mass conservation principles, and assuming thermodynamic equilibrium at all times inside the control volume.

2.3.5 Solar field

In recent years, several studies have underlined the potential of small-capacity thermal solar organic power systems [19, 18, 20]. Different models of parabolic trough solar collector have therefore been added to the library. The large ratio between diameter and length allows a one dimensional discretization of the absorber tube.

The model is composed by two subcomponents, as shown in figure 5: the Flow1Dim component which models the heat transfer fluid flow in the heat collector element (HCE) and the SolAbs component
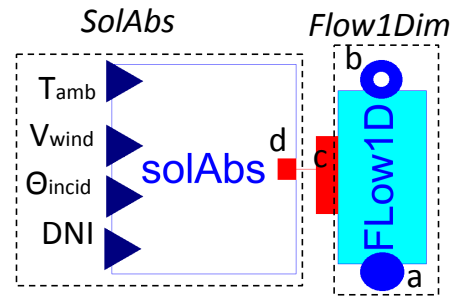

Figure 5: Object diagram of the solar field model. a and b indicate respectively the fluid inlet and outlet ports of the flow model (Flow1Dim). c and d indicate the thermal ports.

which solves the dynamic one dimensional energy balance around the HCE. The solAbs component implements the relations between the environmental parameters, direct normal radiation (DNI), incidence angle (Θ_{incid}), ambient temperature (T_{amb}), wind velocity (v_{wind}) and the axial temperature distribution along the absorber, either based on the Forristal model [12] or based on an empirical correlation derived from the Schott test analysis [3]. The user can choose between different collector geometries based on commercial products. The solar field model is available for compressible and incompressible fluids.

3 The CoolProp2Modelica library

The ThermoCycle library aims at modelling different kinds of thermodynamic cycles, including those with unconventional working fluids. The efficient computation of their thermophysical properties is therefore a key aspect of the proposed models. As previously mentioned, the currently available Modelica solutions for computing organic fluid properties are limited to non-open-source databases. In order to propose a fully open-source tool, ThermoCycle has been coupled to CoolProp, an open-source library developed at the University of Liège [1]. The main features of the CoolProp library are the following:

- Fully open-source

- High accuracy Helmholtz energy-based equations of state.

- More than 110 different working fluids

- Properties over 40 incompressible fluids (e.g. thermal oils) and brines (only accessible with p,T or p,h as input variables).

- Low computational time

The interface between Modelica and CoolProp is implemented in the CoolProp2Modelica library, a modified version of the ExternalMedia library [9].

In thermo-flow systems, the computation of the thermophysical properties amounts for a significant share of the total computational time. In order to enhance fluid property calculation speed, two interpolation methods, the tabular Taylor series expansion (TTSE) and the bicubic interpolation method, have been developed and integrated into the CoolProp main source code [1]. The interpolation tables (the working fluid properties computed on a 200x200 grid over the whole are of states) are built at the beginning of the simulation (at the first property call) and stored in memory for further use. This process requires between 2 s and 10 s. All subsequent property calls are performed using the selected interpolation method. It should be noted that, contrary to the bicubic method, the TTSE method generates discontinuities in the computation of the fluid properties. This discontinuity is however very small and does not constitute an issue during simulation.

In order to assess the effectiveness of the developed methods, a benchmark test was carried out comparing the different available fluid properties database with respect to the elapsed time per property call from Modelica. The properties were called a large amount of times and in different areas of the thermodynamic diagram (liquid, two-phase and vapor). The results are presented in figure 8 and show that Coolprop is significantly faster than Fluidprop and slightly slower than the commercial library TILMedia/RefProp. However, when the TTSE method is activated, the computational time is one order of magnitude lower than TILMedia, demonstrating the efficiency of the proposed interpolation method.
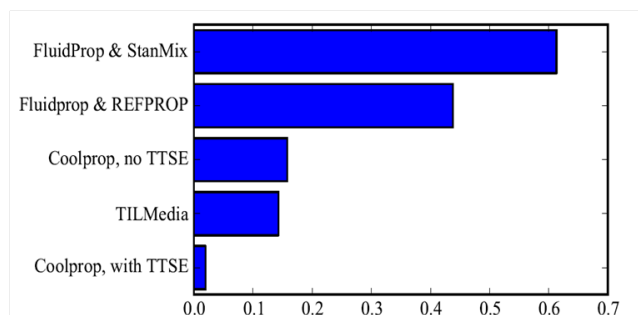

Figure 6: Elapsed time per property call from Modelica in the case of propane (in ms)

In addition to the interpolation methods, smoothing methods to deal with the discontinuity of the density derivatives in the two-phase zone, described in section 4, have also been implemented. The open-source characteristics of CoolProp is a non-negligible advantage since it allows the process to be fully transparent for the end-user, who only needs to activate a flag in the Modelica property call, as shown in figure 7. In this example, the fluid name is modified to instruct Cool-Prop not to compute the transport properties, the debug level is set to unity, the density is smoothed out in the vapor quality interval from 0 to 0.15, and the TTSE interpolation method is activated.

```
package R245faCoolSmooth
 extends CoolProp2Modelica.
 Interfaces.ExternalTwoPhaseMedium(
  mediumName = "R245fa",
  libraryName = "CoolProp",
  substanceNames = {"R245fa|calc_transport=0|debug=1|
  rho_smoothing_xend=0.15|enable_TTSE=1"});
end R245faCoolSmooth;
```

Figure 7: Modelica code to define a fluid. A flag for a numerical and an interpolation method is activated

4 Numerical aspects

Dynamic modelling of thermodynamic cycles can be a challenging task, among others because of the numerous numerical issues arising both during initialization and during integration. In order to enhance the performance and the robustness of the ThermoCycle library, different numerical methods have been implemented and tested. They are presented and discussed in this section.

4.1 Initialization

The convergence of the Newton Solver during initialization is a key challenge when modelling complex system. Several strategies have been developed, such as the *homotopy* method [25]. In addition to this method, a slightly different approach is proposed in ThermoCycle: the system is initialized on a simplified system of equations, and the more complex non-linear equations, such as the computation of the heat transfer coefficients as a function of mass flow, are activated one by one during integration using an *initialization* component developed for that purpose.

4.2 Chattering and flow reversals

The phenomenon of chattering may occur when discontinuities in the model variables are present [15]. This phenomenon can lead to extremely slow simulation, or to simulation failures because the computed variables exceed acceptable boundaries. In discretized two-phase flow models, the main discontinuity often occurs in the density derivative on the liquid saturation curve. Simulation failures or stiff systems can occur if the cell-generated (and purely numerical) flow rate causes a flow reversal in one of the nodes due to this discontinuity. The computation of h_{su} and h_{ex} switches from one value to the other in equations 2 and 3.

In addition to chattering, flow reversals can also result in a singular and non-solvable set of equations, as shown in [24]. A solvability criterion can be expressed as

$$h_{su} > h + \frac{\rho}{\frac{\partial \rho}{\partial h}}, \qquad (7)$$

where $\partial \rho / \partial h$ is a negative term.

This inequality states that in cases of flow reversal, an unsolvable system of equations appears, if the enthalpy of the entering fluid is below a certain limit. A formal demonstration of this effect can be found in [24].

Therefore, to ensure the robustness of the simulation and to avoid chattering or unsolvable systems, two strategies can be employed:

1. Avoid flow reversals caused by the density derivative discontinuity, see Eq. 5.

2. If flow reversal occurs (it is physically possible), make sure that the backward flow enthalpy is higher than the limit described in Eq. 7.

The first strategy can be expressed by an inequality stating that purely numerical, cell-generated flow rates must be lower than the flow rate circulating through the cycle, which can be written (for a single cell):

$$\dot{M}_{ext} \gg \frac{V}{N} \cdot \frac{d\rho}{dt} = \frac{V}{N} \cdot \left(\frac{\partial \rho}{\partial h} \cdot \frac{dh}{dt} + \frac{\partial \rho}{\partial p} \cdot \frac{dp}{dt} \right) \qquad (8)$$

According to Eq. 8, flow reversals and thus chattering or simulation failures are likely to occur if:

- The number of cells (N) is low
- The working fluid flow rate ($\dot{M}_{ext}$) is low
- The internal volume (V) is high
- The working conditions are highly transient (i.e. dp/dt and dh/dt are high)

Different methods are implemented in ThermoCycle to avoid the simulation issues described above. Some are implemented at the Modelica level while others require a modification of the thermodynamic properties of the working fluid and are therefore implemented into CoolProp. It should also be noted that some of these methods have already been proposed in the literature, while some others are new. They are briefly described hereunder. A more comprehensive description is available in [23].

- *Filtering method*: In this strategy, a first order filter is applied to the fast variations of the density with respect to time. This filter therefore acts as "mass damper" and avoids transmitting abrupt variations of the flow rate due the density derivative discontinuity.

- *Truncation method*: This strategy acts on the terms $\partial \rho / \partial p$ and $\partial \rho / \partial h$ of Eq. 8. The peak in the density derivative occurring after the transition from liquid to two-phase is truncated, reducing the numerical flow rate generated by the density derivative discontinuity.

- *Smoothing of the density derivative*: The idea behind this method is to smooth out the density derivative discontinuity using a spline function. Modifications of the thermophysical properties are implemented at the level of the equation of state, i.e. inside the CoolProp database. The main drawback of this method is that the density function is still calculated with the original equation of state: the smoothed density derivative is not consistent with the density function provided by the EOS. This might cause a mass defect during the simulation.

- *Smoothing of the density function*: In order to avoid the mismatch between the density function and its derivative, one possible solution is to smooth the density for a range of vapour qualities (i.e. making it C1-continuous) and recalculating its partial derivatives in the smoothed area. In this situation, the density derivatives are continuous but not smooth, which should still be manageable for the solver.

- *Mean densities method*: The mean densities method was originally proposed by Casella [4] and successfully tested by Bonilla et al. [2]. It is also the method implemented in the ThermoPower Modelica library [7]. A mean density and its partial derivatives are computed in each cell as a function of the node densities, which

eliminates the discontinuity in the partial derivatives.

- *The enthalpy limiter method*: Contrary to the previous methods, the enthalpy limiter method does not aim at avoiding flow reversals. Instead, it ensures that the system of equations remains solvable even in case of flow reversal. As indicated in Eq. 7, the enthalpy of the fluid entering a cell should have a minimum value, ensuring that the system of equations can be solved. The enthalpy limiter method is the practical implementation of this constraint in the cell model. It was originally proposed by Schulze et al. [24] and implemented in the TIL Modelica library.

- *Smooth Reversal Enthalpy*: In case of flow reversal a discontinuity appears in the computed node enthalpy. In this method, this is solved using a smooth transition function for the computation of the enthalpy as a function of the flow rate close to zero. The main drawback is the generation of a highly nonlinear algebraic system that has to be solved by the simulator.

A comparison of these different methods, based on the simulation of a flow model with a high number of cells, in terms of simulation speed, simulation accuracy and mass and energy imbalance shows that adopting the proposed methods can dramatically improve the simulation performance and even allow to simulate flow reversals where traditional models generate simulation failures. The main concern is the error they introduce in the simulation results and the possible mass and energy unbalances they generate. In [23], the methods have been tested on a test system submitted to highly transient conditions. Results have shown that although numerical artifacts are generated, the error remains of the same order of magnitude as the error linked to the standard finite volume model with a 10^{-4} tolerance. This is an important statement showing that the proposed models can ensure failure-free simulation even in highly transient conditions. It should also be noted that those transient conditions are likely to generate chattering or stiff systems are usually concentrated in specific times of the simulation (e.g. start-up and shut-down), in which robustness is more important than accuracy.

5   ThermoCycle Viewer

When debugging or interpreting simulation results, it is generally useful to post-process the raw time-data before displaying it. To that end, an analysis tool has been developed, which scans the simulation results and automatically detects all thermodynamic states to display them on a thermodynamic diagram (e.g. T-s or p-h). The heat exchanger temperature profiles are also detected, displayed, and animated as a function of time.
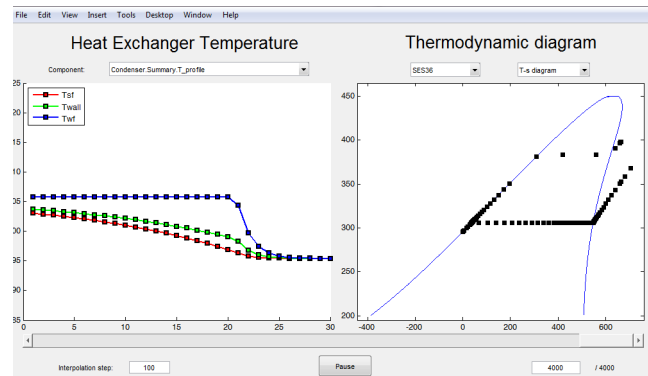

Figure 8: Graphical interface to display thermodynamic states and temperature profiles

6   Conclusion

The development of the ThermoCycle library is an on-going process aiming at providing a completely open-source tool for the dynamic modelling of thermodynamic cycles. The library comprises a number of components which can be used to simulate a wide range of physical systems. Special attention has been paid to the implementation of new and existing numerical methods to enhance the robustness and the simulation speed of the models during initialization and integration. The ThermoCycle and CoolProp2Modelica libraries are being released together with ThermoCycle Viewer through the library website (URL: www.thermocycle.net). Current work now focuses on the integration of new components (moving boundaries models, thermocline and stratified storage, evaporator with frosting, etc.) and on the experimental validation of the proposed models. At the present time, the library and the thermophysical properties database are both open-source, but are running exclusively under Dymola. Porting the library to the OpenModelica platform is another ongoing work, which should lead to a fully integrated open-source solution for the modelling of thermal systems.

7 Acknowledgement

References

[1] Ian H. Bell, Jorrit Wronski, Sylvain Quoilin, and Vincent Lemort. Pure- and Pseudo-Pure Fluid Thermophysical Property Evaluation and the Open-Source Thermophysical Property Library CoolProp. *Industrial & Engineering Chemistry Research*, 2014.

[2] J. Bonilla, L. J. Yebra, and S. Dormido. Mean densities in dynamic mathematical two-phase flow models. *Computer Modeling in Engineering and Sciences (CMES)*, 67(1):13, 2010.

[3] F. Burkholder and C. Kutscher. Heat loss testing of schott's 2008 ptr70 parabolic trough receiver. Technical report, NREL, 2009.

[4] F. Casella. Object-oriented modelling of two-phase fluid flows by the finite volume method. *Proceedings 5th Mathmod Vienna, Austria, Sep*, 2006.

[5] F. Casella, J.G. van Putten, and P. Colonna. Dynamic simulation of a biomass-fired steam power plant: A comparison between causal and a-causal modular modeling IMECE2007-41091. In *Proceedings of the International Mechanical Engineering Congress*, volume 6, pages 205–216, 2007.

[6] Francesco Casella and Alberto Leva. Modelica open library for power plant simulation: design and experimental validation. In *Proceeding of the 2003 Modelica conference, Linkoping, Sweden*, 2003.

[7] Francesco Casella and Alberto Leva. Modelica open library for power plant simulation: design and experimental validation. In *Proceeding of the 2003 Modelica conference, Linkoping, Sweden*, 2003.

[8] Francesco Casella, Tiemo Mathijssen, Piero Colonna, and Jos van Buijtenen. Dynamic modeling of organic rankine cycle power systems. *Journal of Engineering for Gas Turbines and Power*, 135(4):042310, March 2013.

[9] Francesco Casella and Christoph Richter. External media: A library for easy re-use of external fluid property code in madelica. In *In proceeding of the 6th International Modelica Conference*, 2008.

[10] P. Colonna and T. P. van der Stelt. *FluidProp: a program for the estimation of thermo physical properties of fluids*. Energy Technology Section, Delft University of Technology, 2004.

[11] Sebastien Declaye, Sylvain Quoilin, Ludovic Guillaume, and Vincent Lemort. Experimental study on an open-drive scroll expander integrated into an {ORC} (organic rankine cycle) system with {R245fa} as working fluid. *Energy*, 55(0):173 – 183, 2013.

[12] R. Forristall. Heat transfer analysis and modeling of a parabolic trough solar receiver implemented in engineering equation solver. Technical Report Task No. CP032000, National Renewable Energy Laboratory (U.S. Department of Energy), 2003.

[13] Peter Fritzson. *Principles of Object-Oriented Modeling and Simulation with Modelica 2.1*. John Wiley & Sons, August 2010.

[14] L. Guangbin, Z. Yuanyang, L. Yunxia, and L. Liansheng. Simulation of the dynamic processes in a scroll expander-generator used for small-scale organic rankine cycle system. In *Proceedings of the institution of Mechanical Engineers, Part A: Journal of Power and Energy*, 2011.

[15] Jakob Munch Jensen. *Dynamic Modeling of Thermo-fluid Systems: With Focus on Evaporators for Refrigeration : Ph.D.-thesis*. Energy Engineering, Department of Mechanical Engineering, Technical University of Denmark, 2003.

[16] E.W. Lemmon, M.L. Huber, and M.O. McLinden. *NIST Standard Reference Database 23: Reference Fluid Thermodynamic and Transport Properties-REFPROP*. National Institute of Standards and Technology, Boulder, Colorado, 2010.

[17] Vincent Lemort, Sylvain Quoilin, Cristian Cuevas, and Jean Lebrun. Testing and modeling a scroll expander integrated into an organic rankine cycle. *Applied Thermal Engineering*, 29(14-15):3094–3102, 2009.

[18] E. Prabhu. Solar trough organic Rankine electricity system (STORES) stage 1: Power plant optimization and economics. Technical Report NREL/SR-550-39433, National Renewable Energy Laboratory, 2006.

[19] H. Price and V. Hassani. Modular trough power plant cycle and system analysis. Technical Report NREL/TP-550-31240, National Renewable Energy Laboratory, 2002.

[20] S. Quoilin, M. Orosz, H. Hemond, and V. Lemort. Performance and design optimization of a low-cost solar organic rankine cycle for remote power generation. *Journal of Solar Energy Engineering*, 85:955–966, 2011.

[21] Sylvain Quoilin. *Sustainable Energy Conversion Through the Use of Organic Rankine Cycles for Waste Heat Recovery and Solar Applications*. PhD thesis, University of Liege, Belgium, 2011.

[22] Sylvain Quoilin, Richard Aumann, Andreas Grill, Andreas Schuster, Vincent Lemort, and Hartmut Spliethoff. Dynamic modeling and optimal control strategy of waste heat recovery organic rankine cycles. *Applied Energy*, 88(6):2183–2190, 2011.

[23] Sylvain Quoilin, Ian Bell, Adriano Desideri, and Vincent Lemort. Methods to increase the robustness of finite-volume flow models in thermodynamic systems. *Energies*, 2014.

[24] Christian Schulze, Manuel Graber, and Wilhelm Tegethoff. A limiter for preventing singularity in simplified finite volume methods. In *Mathematical Modelling*, 2012.

[25] Michael Sielemann, Francesco Casella, Martin Otter, Christoph Clauß, Jonas Eborn, Sven Erik Mattsson, and Hans Olsson. Robust initialization of differential-algebraic equations using homotopy. In *Proceedings of the 8th Modelica Conference*, pages 21–22, Dresden, 2011.

[26] TLK Thermo GmbH. TIL Suite - Simulates Thermal Systems, April 4th 2013.

[27] Hubertus Tummescheit, Jonas Eborn, and Falko Jens Wagner. Development of a modelica base library for modeling of thermo-hydraulic systems. In *Proceedings of the Modelica Workshop 2000*, 2000.

[28] Jorrit Wronski, Jean-Francois Oudkerk, and Fredrik Haglind. Modelling of a small scale reciprocating ORC expander for cogeneration applications. In *ASME-ORC 2013 – 2nd International Seminar on ORC Power Systems*, 2013.