

Use of External Fluid Property Code in Modelica for Modelling of a Pre-combustion CO₂ Capture Process Involving Multi-Component, Two-Phase Fluids

Carsten Trapp^{*1}, Francesco Casella², Teus van der Stelt¹ and Piero Colonna¹

¹Delft University of Technology, Propulsion and Power, Kluyverweg 1, 2629 HS Delft, The Netherlands

²Politecnico di Milano, Dipartimento di Elettronica, Informazione e Bioingegneria, Via Ponzio 34/5, 20133 Milano, Italy

Abstract

This paper presents the development of a system model for a pre-combustion CO₂ capture process as part of an integrated gasification combined cycle power plant. This process entails the modelling of highly non-ideal, two-phase multi-component mixtures which are currently not supported by available Modelica media libraries or interfaces.

Therefore, an interface prototype was developed and tested for the modelling and simulation of the CO₂ capture process. Limitations concerning the modelling approach and improvements targeting the computational efficiency are discussed. Recommendations about the design of a library for the use of external property estimation code in Modelica conclude the treatment.

Keywords: pre-combustion CO₂ capture; two-phase, multi-component fluids; external fluid property code

1 Introduction

Pre-combustion CO₂ capture applied to integrated gasification combined cycle (IGCC) power plants is a promising technical solution to mitigate CO₂ emissions and therefore the effect of climate change [1]. The integration of the CO₂ removal unit with the very complex gasification process and combined cycle power plant leads to challenges especially regarding dynamic operation. Nowadays, dynamic performance of fossil-fuelled power plants becomes increasingly important as the share of electricity produced by renewable energy sources, which is inherently unsteady, is continuously growing. Therefore, the inte-

grated capture process has to be able to follow frequent and fast load changes to allow for flexible power production. In order to study the transient performance of the pre-combustion CO₂ capture unit during load variations, dynamic models of the entire system and models of the individual components have thus been developed using the Modelica modelling language. The models have been validated by comparison with experimental data obtained from a unique, fully instrumented CO₂ capture pilot plant, which has been realized at the Buggenum IGCC power station in the Netherlands by the utility company Vattenfall [2].

The main challenge of the model development is related to the computation of fluid properties, in particular phase equilibria, due to the fact that highly non-ideal, two-phase, multi-component fluids are involved in the capture process. Currently, Modelica medium models are not available for this type of fluids.

One possibility is to implement required medium models, just as the process models, in the Modelica language, with the advantage to be able to perform efficient simulations as the code can be optimized, provided the equation of state is written in a declarative way, which might not always be possible. However, the implementation of non-ideal fluid property models is rather time-consuming and not trivial, as dedicated solution algorithms might be required for efficiency and numerical robustness.

The other possibility is to make use of available thermophysical property packages and interface these tools with Modelica. Employing external tools for the computation of fluid properties provides some advantages: 1) typically the property software employs dedicated algorithms for fast and robust calculations of the fluid properties, 2) the property package can be interfaced with a wide variety of engineering software tools (e.g. steady-state and dynamic system mod-

^{*}Email: C.Trapp@tudelft.nl, P.Colonna@tudelft.nl

elling, component design, CFD, etc.) allowing for the use of the same thermophysical properties, thus eliminating one common source of uncertainty, 3) a wide range of pure fluids and fluid mixtures described with suitable and accurate equation of states are available in the property package.

This concept was successfully demonstrated with the ExternalMedia library [3] which supports two-phase, single-substance fluids all compatible to the Modelica.Media interface. However, multi-component fluids as required for the CO₂ capture process are not supported because the interface for multi-phase fluid mixtures is limited to two-phase pure (or pseudo-pure) components.

For this reason, a prototype for an interface supporting two-phase, multi-component fluids was developed and tested for the modelling and simulation of the CO₂ capture process.

The objective of the work documented here is to demonstrate the feasibility of modelling such a chemical process with Modelica by making use of external fluid property code and to indicate limitations concerning the modelling approach as well as to discuss possibilities for the improvement of the computational efficiency. Finally, recommendations shall be drawn for the design of a generic interface to external fluid property code.

2 Pre-combustion CO₂ capture process

The simplified process flow diagram of the CO₂ capture pilot plant built at the site of the Buggenum IGCC power station is depicted in Figure 1. The syngas from the gasifier entering the CO₂ removal unit is mixed with process water in order to obtain a pre-set H₂O:CO ratio, and then it is fully evaporated and superheated by means of electrical heaters. The carbon monoxide present in the syngas is converted into hydrogen and carbon dioxide via a three-stage, sweet, high-temperature water-gas shift (WGS) process. The excess process water is recovered from the shifted syngas through condensation and then recycled. Subsequently, the carbon dioxide is removed from the syngas in the CO₂ absorber by means of physical absorption utilizing the solvent dimethylether of polyethylene glycol (DEPEG). The resulting H₂-rich syngas is fed to the gas turbine of the combined cycle power plant and the CO₂ is recovered by three-stage depressurization of the loaded solvent. The lean solvent is recycled to the absorber, while the CO₂-rich product stream is compressed to a state suitable for storage.

Throughout the evaporation and condensation in

the shifting section, vapour-liquid equilibrium of the syngas-water mixture is assumed, which requires rigorous fluid property computations. The water-gas shift reaction occurs at conditions where the syngas-steam mixture can be described as an ideal gas. Throughout the absorption section, vapour-liquid equilibrium of the syngas-solvent mixture is also a verified hypothesis.

The main differences between the pilot plant and a large-scale CO₂ capture process concern the heat integration in the shifting section. In a large-scale design the electrical devices used for heating, cooling and condensation are replaced by heat exchangers.

3 Model development

The thermophysical properties of the two-phase multi-component syngas-water/syngas-solvent mixtures are calculated with the Perturbed Chain - Statistical Associating Fluid Theory (PC-SAFT) equation of state [4] due to its success in predicting vapour/gas-liquid equilibria of complex fluids and mixtures for a broad range of conditions. This EoS has been validated against experimental measurements and data from literature [5] and implemented, together with fast and robust algorithms, into an in-house property package [6] which is interfaced with the dynamic modelling tool via the ModelicaFluidProp Modelica library.

Library architecture

The ModelicaFluidProp library provides the functional interface that allows to integrate external fluid property codes into Modelica models. The library contains two parts, the Modelica front-end which makes various functions available for the calculation of different property sets (for instance "AllProps" or "TwoPhaseDeriv") and the C/C++ back-end, containing C++ objects that carry out the interfacing between the Modelica level and the external software tool.

The Modelica library contains a generic package FluidPropMedium. The actual external fluid property code is specified by setting values to constants such as ModelName, which defines the name of the external library, nComp, which specifies the number of fluid constituents and Comp, which defines the name of the individual constituents. The external medium model can be used in any component model and is not extending any medium package from the standard Modelica library.

The implemented set of functions in the FluidPropMedium package mirrors one-to-one property functions available in the external property tool which

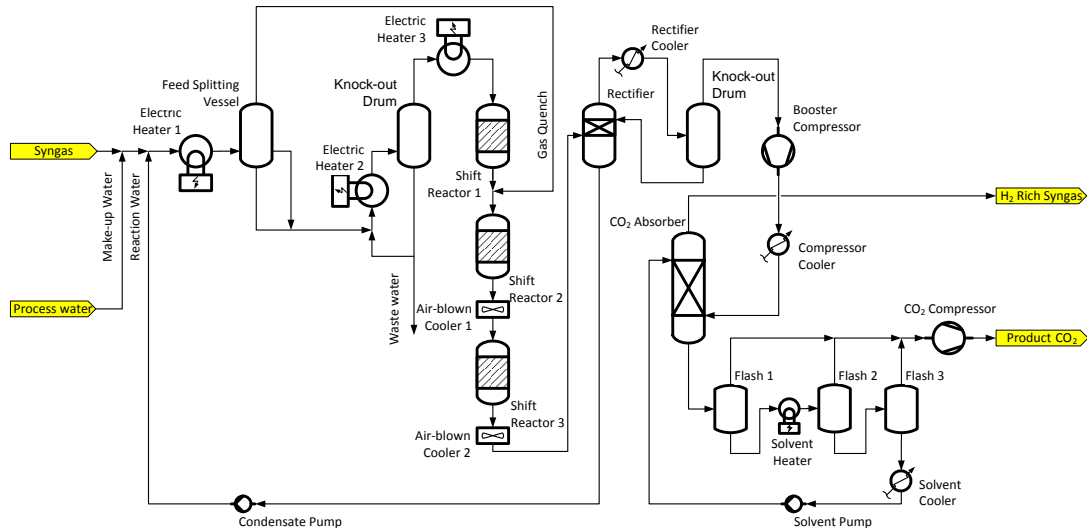


Figure 1: Process flow diagram pilot plant.

are interfaced with corresponding C-functions defined in the C interface layer.

In the following, the working principle of the library is explained, based on the exemplary code below:

```
import SI = Modelica.SIunits;

package SyngasDEPEG
  extends ModelicaFluidProp.FluidPropMedium(
    modelName="PC-SAFT",
    nComp=6,
    Comp={"carbon monoxide","hydrogen",
          "carbon dioxide","water","nitrogen",
          "DEPEG"})
end SyngasDEPEG;

model Example
  SyngasDEPEG.AllPropsOut prop;
  SI.Temperature T;
  SI.Pressure P;
  SI.SpecificEnthalpy h;
  SI.Density d;
  SI.MoleFraction Y[SyngasDEPEG.nComp]=
  SyngasDEPEG.reference_X;
  SI.MoleFraction Yliq[nComp]
  "Molar fractions of liquid phase";
  SI.MoleFraction Yvap[nComp]
  "Molar fractions of vapour phase";
equation
  (prop,Yliq,Yvap) =
  Medium.AllProps("PT",P,T,Y);
  h = prop.h;
  d = prop.d;
end Example;
```

The AllProps function of the FluidPropMedium package calls the corresponding C function of the interface and passes the specification of the thermodynamic state ("PT"), the values of pressure, temperature and composition as well as the constants for medium identification. The interface function handles the creation of an object for the external property code and the execution of the solver to compute

the required properties. The calculated fluid properties are passed via the prop record to the Modelica code. The AllProps function returns all primary thermodynamic properties such as P , T , v , d , h , s , u , etc. which can be computed with hardly any additional computational cost when solving the equation of state. Secondary thermodynamic properties such as heat capacity, speed of sound, various single-phase partial derivatives and transport properties are computed with a separate function as these properties are less often needed and require additional computations. The computationally expensive two-phase partial derivatives are combined in another property function.

The arrangement of primary and secondary fluid properties in meaningful functions allows for a flexible use and avoids unnecessary repeated computations.

Process models

The objective is to develop physical-based component models which allow the modelling and simulation of the pre-combustion CO₂ capture process. The model structure shall facilitate the integration of external functions for thermophysical property calculations.

For the modelling of the CO₂ capture process various component models are required. Whenever possible models were reused from available Modelica libraries. For example, basic component models such as sinks, sources, valves, pressure drops, pumps, heat exchange and flow models, are taken from the ThermoPower library [7, 8] and adapted in terms of their media models which have been replaced with functional calls to the external property tool.

For the following components new models were implemented:

- **Flash vessel**

The process of phase separation is modelled under the assumption of thermodynamic equilibrium between the liquid and vapour phase at all times. This is a justified assumption since the process is typically designed to ensure sufficient mixing. The model describes the holdup of vapour and liquid with conservation equations which account for both phases together. Saturated conditions are assumed for the liquid and vapour outlet streams and therefore entrainment of liquid in the vapour flow is neglected. The flash vessel model is implemented as a "pure storage" component and hence frictional losses are not considered. The static pressure head due to the liquid level in the vessel is accounted for in the algebraic momentum balance. Heat transfer from the fluid (both vapour and liquid phase) to the vessel wall and accumulation of thermal energy in the wall as well as heat losses to the environment are neglected. Thus, also superficial condensation is assumed to be negligible.

- **Water-gas shift reactor**

The reaction of carbon monoxide with steam to produce carbon dioxide and hydrogen is described in a lumped-parameter model. The syngas entering and leaving the reactor is an ideal-gas mixture containing CO, CO₂, H₂, H₂O, N₂. Other trace constituents are neglected. The model accounts only for the WGS reaction. Intermediate reactions involving other chemical species are neglected. The reactor model is subdivided into five sub-models (Figure 2): reaction node, mixing gas volume, convective heat transfer, thermal storage and pressure drop.

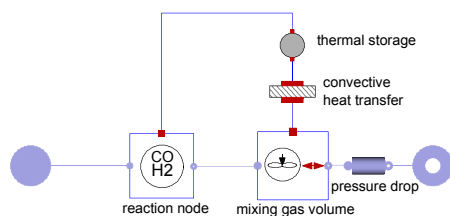


Figure 2: Object diagram of reactor component.

The WGS reaction takes place in an infinitesimally small volume (reaction node) representing one finite discretization of the catalyst and reaches thermodynamic equilibrium. The accumulation of mass and energy in the bulk phase of

the reactor are described in a perfectly mixed volume (mixing gas volume) which receives the reaction products. This control volume exchanges heat with the catalyst by means of convection. The storage model describes the accumulation of thermal energy in the catalyst. Heat transfer to the environment is neglected.

The water-gas shift reactor is discretized in axial direction by an array of reactor models in order to correctly describe the gradual changes in reactor outlet conditions during transient operation. Changes in the reactor inlet conditions reach the reactor outlet with a delay due to thermal storage in the catalyst, which cannot be represented with a 0-dimensional model due to the high number of transfer units between the gas and the catalyst itself.

However, this one-dimensional discretization does not represent the the actual axial reactor profile as equilibrium conditions are assumed in each reactor model element for simplicity. In steady-state conditions the equilibrium temperature is reached at each discretization of the catalyst, which also determines the temperature dependent WGS reaction.

- **Pilot plant specific heater and cooler components**

Various electrical components for evaporation, superheating, cooling and condensation were in particular developed for the pilot plant process and will not be part of a large-scale plant in this specific configuration. The models were developed following a modular approach such that sub-models can be reused. The models were typically subdivided, if applicable, in flow models, heat transfer models and thermal storage models. Whenever possible models from the ThermoPower library were used, typically in case the medium was water or ideal gas, or adapted.

- **Absorption column**

The packed column model for physical absorption (no chemical reactions) is discretized in theoretical stages in axial direction and counter-current flow of the vapour and liquid is assumed. Each stage is modelled by an equivalent tray module ("pure storage") and a resistive module. By connecting a series of storage and resistive modules (Figure 3) a low index of the equation system can be maintained (detailed discussion see Section 4).

In the equivalent tray module, pressure, temperature and composition of the liquid and vapour

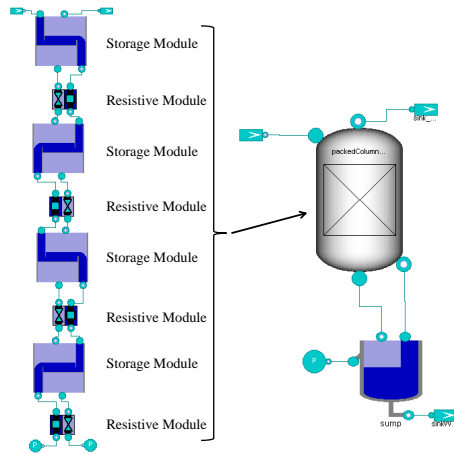


Figure 3: Model structure of the absorption column.

phase are determined by solving the conservation equations for mass and energy assuming thermodynamic equilibrium between liquid and vapour [9]. This module is based on the flash vessel model with its assumptions stated above. In the resistive module the momentum equation is substituted by empirical correlations to describe the hydrodynamics of the stage predicting the liquid and vapour flow rate as a function of the pressure difference between the stages, the liquid holdup and the packing characteristics. Empirical correlations for the pressure drop and liquid holdup are published by Stichlmair et al. [10] and Billet and Schultes [11].

4 Modelling approach / Lessons learned

The use of external fluid property functions in Modelica process models puts some restrictions on the model development. Specific attention requires the formulation of the differential model equations, the choice of state variables and the causality of the system model. These issues are addressed in the following.

4.1 Choice of state variables

For dynamic modelling of thermo-physical systems not only the choice of the system state variables is of importance but also the selection of the thermodynamic states used to determine fluid properties. In general, the system state variables should allow for an easy computation of the thermodynamic properties required to determine the system performance and the thermodynamic state variables should unambiguously determine the fluid state. Further, the choice of sys-

tem state variables can have a significant influence on ease of initialization, numerical robustness and computational speed.

In the following, three different possibilities for the choice of state variables and their influence on resulting differential and algebraic equation (DAE) system as well as simulations speed are analysed. As an example, dynamic mass and energy balances for describing storage of vapour and liquid in a volume under the assumption of thermodynamic equilibrium are used. These equations can be found in the flash vessel and absorber tray models.

Explicit system state variables M, u, X_i

In the most simple way the dynamic mass and energy balance can be formulated as

$$\frac{dM}{dt} X_i + M \frac{dX_i}{dt} = w_{in} X_{in,i} - w_{liq} X_{liq,i} - w_{vap} X_{vap,i}, \quad (1)$$

$$M \frac{du}{dt} + u \frac{dM}{dt} = w_{in} h_{in} - w_{liq} h_{liq} - w_{vap} h_{vap}, \quad (2)$$

where M is the total mass, u is the internal energy, X_i is the component mass fraction vector whereby the subscript i ranges from 1 to the number of species in the mixture, w is the mass flow rate and h is the specific enthalpy. A formulation purely based on moles is also possible and might be the preferred choice as conversions between mole and mass are avoided.

Assuming M, u and X_i are selected as state variables and no other variables than the states appear under the time derivative the system solution can be obtained straightforward. This applies, for example, to a single flash vessel model where the index of the DAE system is 1. Section 4.2 covers the case of higher index problems, e.g. when dealing with a model of two flash vessels connected with a zero pressure drop.

The mass and energy equations represent a compact and very declarative formulation of the conservation equations. The pressure P , the temperature T and the mass composition X_i are chosen as thermodynamic states, therefore

$$prop = prop(P, T, X_i). \quad (3)$$

For this state selection property computations for two-phase mixtures are most efficient and robust with the used external media library (see Section 4.3). As the system states and thermodynamic states differ, implicit equations must be solved at each time step in order to determine the thermodynamic state variables.

Consequently, with the solution strategy used by Dymola the computational speed is affected by the iterations on the property function, which are anyway the most expensive part of the entire simulation.

Implicit system state variables P, T, X_i

The Modelica language features the option to change states by means of the StateSelect attribute, without changing the declarative formulation of the dynamic mass and energy balance. Hence, pressure, temperature and composition can be selected as preferred states in accordance to the thermodynamic states with the aim to reduce iterations during the solution procedure. However, as the total mass and internal energy remain under the time derivative and not being states, the tool attempts to symbolically differentiate M and u with respect to the states P, T, X_i in order to establish a relationship between the old states, which have been discarded, and the newly selected states. This procedure fails as the external property call cannot be differentiated.

Explicit system state variables P, T, X_i

In order to facilitate a feasible solution an explicit expression for $\frac{dM}{dt}$ and $\frac{du}{dt}$ needs to be provided as a function of the state variable derivatives $\frac{dP}{dt}$, $\frac{dT}{dt}$ and $\frac{dX_i}{dt}$. The needed time derivatives therefore read

$$\frac{dM}{dt} = -Md \left[\left(\frac{\partial v}{\partial p} \right)_{T,X} \frac{dp}{dt} + \left(\frac{\partial v}{\partial T} \right)_{p,X} \frac{dT}{dt} + \left(\frac{\partial v}{\partial X_i} \right)_{p,T,X_{j \neq i}} \frac{dX_i}{dt} \right], \quad (4)$$

$$\frac{du}{dt} = \left(\frac{\partial u}{\partial p} \right)_{T,X} \frac{dp}{dt} + \left(\frac{\partial u}{\partial T} \right)_{p,X} \frac{dT}{dt} + \left(\frac{\partial u}{\partial X_i} \right)_{p,T,X_{j \neq i}} \frac{dX_i}{dt}. \quad (5)$$

The partial derivatives of u and v with respect to P, T, X_i need to be provided by the external media library. Two-phase partial derivatives of mixtures, which are based on properties obtained from the phase equilibrium calculations, are commonly not available among the typical thermophysical fluid properties and therefore have been specifically implemented in the framework of this model development. These mixture derivatives are currently computed numerically. Future work might consider the analytical formulation as implemented, for example, in Multiflash [12]. However, also the calculations based on analytical expressions is expected to be computational expensive [13].

These mixture derivatives cannot be fully expressed analytically and hence the numerical calculation affects the computational speed of the dynamic model.

In Figure 4 the simulation performance of a flash vessel with system states M, u, X_i and state variables P, T, X_i is compared. The simulation of the latter model requires 90 % less functional calls to the external property library promoted by the same choice for system and thermodynamic state variables. However, considering the computational time the latter model only leads to an improvement of 40 %. This is explained by the additional calculation of partial derivatives required when changing the states from M, u, X_i to P, T, X_i .

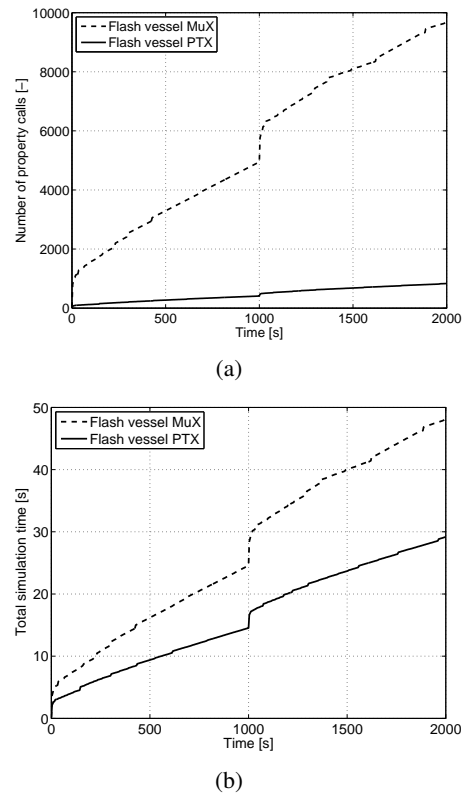


Figure 4: Comparison of flash vessel models with state variables M, u, X_i and P, T, X_i . a) Number of property calls. b) Total simulation time.

The observed difference in computational speed between both choices of state variables might also be related to the solution strategy employed by the software tool. Dymola was adopted for this project, and Dymola obtains the solution of the DAE system by iterating on a nested loop, and solving the ODE's in the outer and the algebraic equations in the inner loop. It might be possible that using a DAE solver directly on the original DAE system in this case might lead to a much smaller difference in computational speed between both solutions. However, a direct solution strat-

egy might entail more trouble during initialization and shorter time steps to achieve convergence might be required. A hybrid strategy where both approaches are applied, dynamically switched, might be interesting to explore.

4.2 Developing index-1 models

Considering a model containing two flash vessels, component mass and energy balance represented by equation 1 and 2, with zero or constant pressure drop, then the index of the DAE system is larger than 1, assuming M , u and X_i are chosen as state variables.

The solution of a DAE system with higher index is commonly obtained by symbolic manipulation of the equations system in order to reduce its index to 1. Current simulation tools implementing the Modelica language employ state-of-the-art techniques for index reduction. Difficulties during index reduction might arise in case fluid property calculations must be symbolically differentiated. If the fluid correlations or equation of state are implemented as a Modelica media library, possibly accompanied with annotations to compute its time derivatives, symbolic manipulation can be performed by the tool resulting in a successful index reduction. However, in case external media libraries are used, interfaced with the Modelica models, index reduction fails as external functions cannot be manipulated. This can be resolved by either supplying any time derivatives required for index reduction or by developing the Modelica models such that the DAE system remains in the index-1 form. The latter approach has been followed for the dynamic modelling of the CO₂ capture pilot.

Causal versus acausal approach

Following the acausal modelling approach, which is fully supported by Modelica, connections of sub-models might not respect the causality leading to a system of DAE's of higher index (example of two flash vessels described above). Therefore, a mixed approach is applied during the model development. The sub-models and interfaces are defined and developed in an object-oriented manner such that the models can be employed in an acausal context. However, the system has been analysed and decomposed into subsystems following the causal modelling paradigm in order to avoid the occurrence of higher order DAE systems [14]:

- Identification of bilaterally coupled variables of the models.
- Discretisation of the model in resistive and storage modules, namely solving the conservation

laws for flow and potential in different control volumes.

- Connect the resistive to the storage modules and vice versa such that potential variables are inputs of the resistive and outputs of the storage modules. Flow variables are inputs to the storage and outputs to the resistive modules.

By following a more causal development and arrangement of the models the DAE system can be maintained in the index-1 form which allows for a straightforward use of external property functions. The disadvantage of this approach is that it poses restrictions on the connection of modules. It might be necessary to include dummy modules with no process functionality in order to maintain the resistive-storage structure.

4.3 Improvement of computational time

Based on the current experience obtained from the modelling of the CO₂ capture process, the computation of thermodynamic fluid properties, in particular phase equilibria, accounts for the main share of the simulation time (in the order of 95 %). In addition, much more complex and hence computational expensive thermodynamic models are required when dealing with highly non-ideal, multi-component mixtures in comparison to fluid models for water or ideal gas. Therefore, an appropriate and smart use of property functions should be considered during the entire model development process. In the following, suggestions are presented, which might significantly contribute to a successful convergence and more efficient simulations.

Choice of thermodynamic states

For an efficient use of external property functions in Modelica, it is necessary to have knowledge on the property calculations performed in the external tool. The thermodynamic state can be determined for different choices of independent variables, for example "PT", "Pv", "Ph", "Pq", "Tv", "Tq", "uv", etc., where P is the pressure, T is the temperature, v the specific volume, h is the specific enthalpy, q is the vapour quality and u is the specific internal energy. In case of mixtures the component fraction vector X_i is added to set of independent variables.

The external tool employs an isothermal ("PT") or an isenthalpic ("Ph") flash algorithm developed by Michelsen [15] for the PC-SAFT and cubic EoS's, which are robust and reliable equilibrium calculations based on the minimization of Gibbs free energy. If other thermodynamic states are chosen as input, then

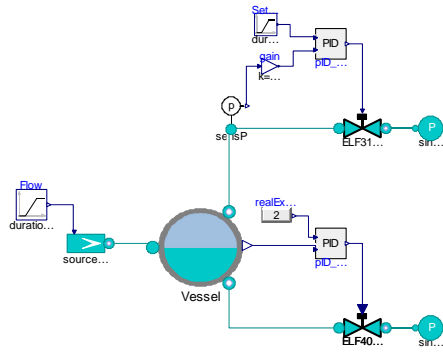


Figure 5: Object diagram of a flash vessel with vapour and liquid control valves.

the solution is obtained by iteration on either the PT- or Ph-flash calculation. Any computations including the vapour quality q as input use a bubble/dew point calculation, which is much more difficult to perform and is far less robust than the flash algorithms.

To conclude, "PT" or "Ph" are recommended as thermodynamic states as these inputs allow for fast and robust fluid property calculations.

Single and two-phase property calculations

The flash algorithm first determines the vapour and liquid composition of the fluid (computation of phase equilibrium), which indicates if the fluid state is in the two-phase or the single phase region. Then the vapour and liquid properties such as v , h , s , and u are computed. If applicable two-phase properties are calculated based on the single phase properties using appropriate mixing rules. The initial determination of the liquid and vapour composition is computationally very expensive and hence this step should be omitted if it is known a-prior that the fluid is present in the single phase. Therefore, the option of skipping the flash calculation and just performing a single phase calculation was implemented by using, for example, "PT-1ph" as input specification.

The flash vessel component is one example where this ability to explicitly indicate the fluid phase finds application. For the conditions in the vessel two-phase properties are required whereas in valves connected to the vapour and liquid outlet single phase properties are sufficient. Experience has shown, that by using single phase property computations where applicable throughout the process the simulation time can be reduced significantly (see Table 1).

Redundant property calls

The model of a simple phase separation as depicted in Figure 5 provides another example on how the com-

putational efficiency can be improved by optimizing the property calculations. The flash vessel model contains a property call which determines required primary two-phase, liquid and vapour properties, such as h_{2ph} , d_{2ph} , h_{liq} , d_{liq} , h_{vap} and d_{vap} ,

$$prop_{2ph,liq,vap} = prop(P, T, X_{i,2ph}). \quad (6)$$

In the valve model connected to the vapour outlet of the flash vessel, the vapour density d_{vap} is required to close the set of conservation equations by

$$prop_{vap} = prop(P, T, X_{i,vap}). \quad (7)$$

It is obvious that, under the assumption of adiabatic operation and no frictional losses, both property computations provide the same result for the vapour density. However, the property calls are different due to the fact that in the vessel the two-phase mixture composition and in the valve the vapour composition is used as input. During the process of translating the model into a set of solvable equations, the compiler will not realize that the second property call is in principle redundant.

One solution to overcome this issue is to transfer the required density via additional output and input connectors from the flash vessel to the valve. This solution has been implemented as optional choice, which can be activated by the modeller. Another possibility would be the use of conditional connectors which transfer next to the flow and potential variables possibly an array of all fluid properties.

The improvement in computational time is summarized in Table 1. In comparison to the use of single phase property calls in the vapour valve the improvement is rather small. However, the aim is not to obtain the most efficient simulations but to demonstrate a modelling approach which might contribute to efficiency.

Approximation of two-phase partial derivatives

One of the most time-consuming part of the property calculations is the computation of two-phase partial derivatives, which are for example required in the flash vessel component and absorber tray model (see equations 4 and 5)

The results of the partial derivatives are used in the mass and energy balances. Various tests indicated that if the change in the absolute value of the thermodynamic states during the simulations is small the resulting change in the value of the partial derivatives has hardly any impact on the simulation results. Therefore, following procedure has been implemented for the models of the CO₂ capture process.

Table 1: Simulation results of model with flash vessel and two control valves simulated for 2000 seconds.

Case	Valves	Vessel	Vessel	Total time [s]
	AllProps [s]	AllProps [s]	TwoPhaseDeriv [s]	
Reference (Valves normal "PT")	107.6	5.2	21.5	134.3
Valves "PT_1ph"	11.2	5.2	21.5	37.9
Vapour density transfer	7.2	5.2	21.5	33.9
Valves "PT_1ph", CallID	11.1	5.2	4.5	20.8

An additional variable (CallID) has been added to each two-phase component assigning each partial derivative function a unique identification. With this identification it is possible to distinguishably store simulation results in the Modelica-FluidProp interface of the different two-phase derivative calls throughout the modelled process (eg. LP Vessel CallID=1, MP Vessel CallID=2,...). Assuming the derivative function with the ID 1 is executed at a certain simulation time instance, then a check is performed if the difference between the current and the values from the previous time instance (which have been stored in the interface) are below a defined threshold. If that is the case, no property computation for the partial derivatives is performed but the stored results of the derivatives from the previous time instance are returned straight to Modelica. In case the threshold is exceeded, then a normal property calculation is performed with the external tool and the previous results stored in the interface are overwritten. This procedure has the benefit that computational time is saved if the change in the absolute value of the partial derivatives is marginal and thus has no impact on the solution. When modelling complex processes involving two-phase multi-component fluids a significant reduction in computational time can be obtained. Exemplary a comparison is provided for the model of the flash vessel with two control valves (see Table 1). The time spent for derivative computations reduces from 21.5 to 4.5 seconds for such a simple model.

5 Recommendations for a future interface

The presented model development approach leads to solvable models by making a smart choice for the thermodynamic and system state variables and by manually applying measures in order to keep the system in index-1 form. However, for reasons of numerical robustness, simulation speed and ease of initialization, a different choice of state variables might be more convenient than the one where the differentiated variables

are used as states. In order to allow for flexible state variable change and automatic index reduction partial derivatives of the thermodynamic properties are essential when using external tools, as demonstrated in this paper.

The goal is to design a Modelica library that interfaces to external property packages, whereby Modelica tools can automatically compute the total time derivatives of each variables in a set A (e.g. density, specific energy, specific enthalpy, ...) with respect to any meaningful subset of variables in a set B (e.g. pressure, temperature, specific enthalpy, ...) that uniquely identifies the thermodynamic properties of the fluid, including multi-component fluids and two-phase mixtures. This is required to successfully carry out the index reduction and/or state variable change task automatically. A first attempt on how to perform automated state variable change is presented by Wellner et al. [16].

At a higher level, this requires setting up a Modelica infrastructure where annotations point to the appropriate functions to compute all the required derivatives. At a lower level, it has to be ensured that the external property package can compute all required derivatives efficiently, i.e., by avoiding unnecessary duplicate computations.

6 Conclusions

This paper presents the development of a system model for a pre-combustion CO₂ capture process as part of an integrated gasification combined cycle power plant, which entails the modelling of highly non-ideal, two-phase multi-component mixtures. As this type of mixtures are currently not supported by available Modelica media libraries, an interface prototype was developed and tested with the fluid property package FluidProp for the modelling and simulation of the CO₂ capture process. Due to limitations regarding index reduction if using external property functions, an approach on how to develop index-1 models and choose system state variables as well as thermodynamic states appropriately is discussed.

For this type of simulations it appears that the computation of thermodynamic properties, in particular phase equilibria, accounts for the main share of the simulation time, therefore various ideas (single versus two-phase property calculations, decrease in redundancy, approximation of partial derivatives) are presented targeting computational efficiency. Further developments might focus on facilitating automated index reduction and making a wide range of partial derivatives available in a flexible and efficient manner. In an ideal setting, the tool allows for different choices of state variables and different solution strategies for the DAE system to find the best combination for the specific case of study.

Acknowledgements

The work documented in this paper has been performed within the CO₂ Catch-up R&D programme aimed at demonstrating and optimising pre-combustion CO₂ capture technology for the energy sector. This programme is executed in a consortium of Vattenfall, TU Delft and ECN.

References

- [1] K. Damen, M. Troost, A. Faaij, W. Turkenburg, A comparison of electricity and hydrogen production systems with CO₂ capture and storage. part a: Review and selection of promising conversion and capture technologies, *Progress in Energy and Combustion Science* 32 (2) (2006) 215–246.
- [2] K. Damen, R. Gnutek, J. Kaptein, N. Nannan, B. Oyarzun, C. Trapp, P. Colonna, E. van Dijk, J. Gross, A. Bardow, Developments in the pre-combustion CO₂ capture pilot plant at the Buggenum IGCC, *Energy Procedia* 4 (2011) 874–881.
- [3] F. Casella, C. C. Richter, ExternalMedia: a Library for Easy Re-Use of External Fluid Property Code in Modelica, in: *Proceedings 6th International Modelica Conference*, 2008, pp. 157–161.
- [4] J. Gross, G. Sadowski, Perturbed-Chain SAFT: An Equation of State Based on a Perturbation Theory for Chain Molecules, *Ind. Eng. Chem. Res.* 40 (2001) 1244–1260.
- [5] N. R. Nannan, C. M. D. Servi, T. van der Stelt, P. Colonna, , A. Bardow, An equation of state based on pc-saft for physical solvents composed of polyethylene glycol dimethylethers, *Industrial and Engineering Chemistry Research* 52 (2013) 18401–18412.
- [6] P. Colonna, T. van der Stelt, A. Guardone, Fluid-Prop: a program for the estimation of thermo-physical properties of fluids. Version 2.4, software (2004-2012).
URL <http://www.FluidProp.com>
- [7] F. Casella, A. Leva, Modelling of thermo-hydraulic power generation processes using modelica, *Math. Comput. Model. Dyn. Syst.* 12(1) (2006) 19–33.
- [8] F. Casella, A. Leva, Object-oriented modelling & simulation of power plants with modelica, pp. 7597–7602, in *Proceedings 44th IEEE Conference on Decision and Control and European Control Conference 2005*, Seville, Spain, Dec. 12-15, 2005.
- [9] J. D. Seader, E. J. Henley, *Separation Process Principles*, Wiley, 1998.
- [10] J. Stichlmair, J. L. Bravo, J. Fair, General model for prediction of pressure drop and capacity of countercurrent gas/liquid packed columns, *Gas Separation & Purification* 3 (1989) 19–28.
- [11] R. Billet, M. Schultes, Prediction of mass transfer columns with dumped and arranged packings: Updated summary of the calculation method of Billet and Schultes, *Chemical Engineering Research and Design* 77, Issue 6 (1999) 498–504.
- [12] Infochem Computer Services Ltd, Multiflash - Flow Assurance Software, www.kbc.com (2014).
- [13] Richard Szczepanski, Infochem Computer Services Ltd, Private communication (2014).
- [14] F. Casella, J. Van Putten, P. Colonna, Dynamic simulation of a biomass-fired steam power plant: a comparison between causal and a-causal modular modeling, *ASME International Mechanical Engineering Congress and Exposition*, *Proceedings* 6 (2008) 205–216.
- [15] M. L. Michelsen, J. M. Mollerup, *Thermodynamic Models: Fundamentals & computational aspects*, second edition Edition, Tie-Line Publications, 2007.
- [16] K. Wellner, C. Trapp, G. Schmitz, F. Casella, *Interfacing Models for Thermal Separation Processes with Fluid Property Data from External Sources*, submitted to the 10th Modelica Conference, Lund, Sweden (March 10-12, 2014).