Optimization-friendly thermodynamic properties of water and steam

Marcus Åberg[1] Johan Windahl[2] Håkan Runvik[2] Fredrik Magnusson[1]

[1]Department of Automatic Control, Lund University, Sweden, {marcus.aberg@gmail.com, fredrik.magnusson@control.lth}

[2]Modelon AB, Ideon Science Park, Lund, Sweden, {johan.windahl@modelon.com, hakan.runvik@modelon.com}

Abstract

This paper describes the development of an optimization-friendly thermodynamic property model of water and steam that covers liquid, vapor, 2-phase as well as the super-critical region. All equations are at least twice continuously differentiable with respect to all model variables and can be used in dynamic optimization problems solved by efficient derivative-based algorithms. The accuracy has been verified against the industry standard IAPWS IF97 and performance and robustness have been tested by solving a trajectory optimization problem where the start-up time of a gas power plant has been minimized while satisfying constraints on temperature gradients, pressure and flows. Simulations of various plant models have also been performed to verify and benchmark the implementation. The results show that the new media can be used in both solving dynamic optimization and simulation problems yielding reliable results. The new media has been integrated into Modelon's Thermal Power library 1.13. This article is built upon the work in *(Åberg, 2016)*.

Keywords: Dynamic optimization, Thermodynamic properties, Power plant start-up, ThermalPower library, WaterIF97, Optimica, JModelica.org

1 Introduction

During the last decade, optimization of large scale dynamical systems has become more common in both the industry as well as in academia (Magnusson, 2016). There are several interesting areas and applications where optimization can be used, e.g. to improve efficiency and economical aspects in energy applications. Examples where Modelica models have been used include start-up of power plants (Casella, Donida, & Åkesson, 2011), (Runvik, 2014), (Parini, 2015), production planning of district heating networks (Velut, et al., 2014) and power plant load scheduling (Kumar & Mathur, 2014). Modelica is well suited to describe the behavior of dynamical models and thereby also suitable to be used in the context of optimization.

Even if the usage is more common today, the use of dynamic optimization is still not widely spread among the engineering community as compared to simulation. There are several factors that have been limiting the deployment:

- Modelica does not support formulation of optimization problems. However, it can easily be formulated using the Modelica extension Optimica (Åkesson, 2008) or using custom annotations (Zimmer, Otter, Elmqvist, & Kurzbach, 2014)
- It is more challenging to create optimization models versus simulation models. Solving efficiently large-scale dynamical optimization problems requires the model equations to be at least twice continuously differentiable. In the general case when solving non-convex dynamic optimization problems good initial guess values, appropriate model dynamics and as well as good numerical properties are required to find the optimal solution (Nocedel & Wright, 2006)
- Modelica libraries such as the Modelica standard library have been designed for simulation and not optimization. The lack of libraries for optimization is usually a stopper as creating robust models is a large effort and requires an understanding of numerical aspects.

This work targets the last issue and is intended as a first step to bridge the gap between simulation and optimization of thermo-fluid systems. We do so by implementing an optimization-friendly water and steam property model that fulfills a generic media interface.

Modelica is object oriented and supports design of interfaces and classes. This allows a library designer to create models of various fidelity and assumptions. Users can then change between classes that fulfill the constraining interface. Examples include switching to a media of lower fidelity that is less computationally demanding for e.g. real-time applications or to a model suitable for optimization.

The choice of focus on water and steam properties is due to its large usage in power and heat applications. Traditional electricity-generation sources such as coal, nuclear and natural combined gas plants are based on a steam-cycle. Other applications are hydro power plants and heating and cooling distribution networks.

2 Background

The availability of a Modelica implementation of the industry standard of water and steam properties IF97 (Wagner, et al., 2000) helped to spread the usage of the Modelica technology to the energy and power sector (Windahl, et al., 2014). But the high accuracy implementation Modelica.Media.WaterIF97 is targeting the usage of simulation and not optimization. The main issues with using Modelica.Media.WaterIF97 for optimization are:

- limited support of first order and no support of second order partial derivatives of thermodynamic properties
- discontinuous first order partial derivatives at the phase borders between liquid and steam
- discontinuous first order partial derivatives at the region boundaries. IF97 is divided into 5 regions that have their own implementation (Wagner, et al., 2000)

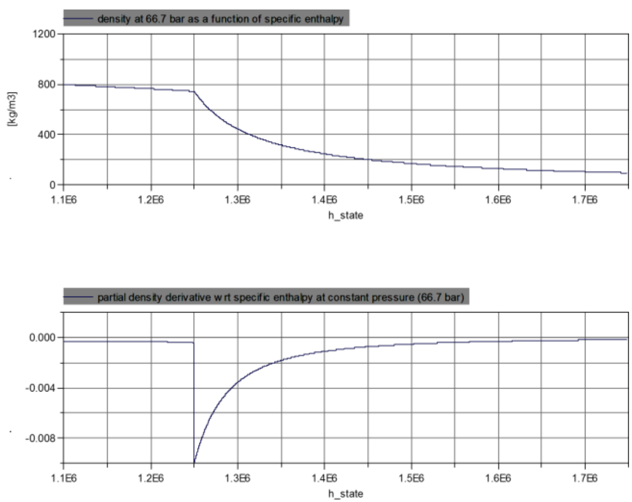

Figure 1 Density (upper) and its partial derivative with respect to specific enthalpy at constant pressure as a function of specific enthalpy. At h=1250 kJ/kg is the bubble saturation line for water which introduces a discontinuity in the partial derivative.

The lack of support of derivatives is an implementation issue. Modelon.Media.WaterIF97, a similar implementation, has support for first order derivatives. But the discontinuity at the phase regions, as illustrated in Figure 1, is a fundamental limitation. The formation or depletion of a phase is a strong non-linear process and needs to be approximated to be twice continuously differentiable. The models in this work are implemented to be compatible with JModelica.org's dynamic optimization framework (Magnusson & Åkesson, 2015). This framework uses CasADi (Andersson, 2013), to efficiently compute sparse first and second order derivatives using algorithmic differentiation (Griewank & Walther, 2008).

2.1 Previous work

To the authors knowledge there is no published work related to dynamic optimization of energy and power systems that focus on a generic media implementation. (Velut, et al., 2014) and (Runvik, 2014) mention the use of “smooth media model functions” but don’t go into any detail. (Casella, Donida, & Åkesson, 2011) use simplifications such as incompressible fluids with constant heat capacity for non-saturated liquid and steam. (Parini, 2015) approximates the subcooled liquid as incompressible fluid and describe the superheated vapor using a cubic equation of state but does not describe any accuracy or region of validity. (Windahl, et al., 2014) investigate requirements for a new media interface, mentioning the benefit of an interface that supports analytic calculations of the Hessian but don’t go any further. (Schulze, 2014) focuses on numerically efficient implementation but does so from a simulation perspective. This is also the focus of the guideline on the fast calculation of steam and water properties with the spline-based table look-up method (International Association for the Properties of Water and Steam, 2015)). The latter uses quadratic splines that are continuously differentiable once and therefore not suitable for dynamic optimization.

This article is built upon the work in (Åberg, 2016). To this publication the media implementation has been updated with some minor modifications that have made the model more numerically efficient compared to the implementation used in that thesis. Therefore the results in this article have been updated too.

3 Implementation

The approach chosen was to approximate the thermodynamical functions with polynomials over different operating regions in the p-h, p-s, p-T and d-T plane. These approximations are then connected via smooth step functions from one region to another. In that way, the functions are twice continuously differentiable over the whole working regime.

DOI 10.3384/ecp17132449

Polynomial approximation over different regions has the main advantage that it is smooth over the defined region. The main challenge here is to find a way to accurately and smoothly connect the different regions. In this implementation step functions are used to make a smooth transition between the regions. These can be defined so that the functions are twice continuously differentiable and the smoothness requirement hence is fulfilled.

The functions that were implemented can be divided into 1D and 2D-functions. 1D-functions describe the saturated behavior in the two-phase region. Saturation temperature, bubble and dew enthalpy are quantities that can be calculated directly from the pressure. The 2D-functions take two independent state properties (Thorade & Saadat, 2013) and calculate thermodynamic properties and a few partial derivatives.

The methods of least squares are used to fit a univariate or bivariate polynomial to the specified data set. The maximum order of the polynomials was set to $k = 9$ on following form.

$$p(x_1, x_2) = \sum_{i=0}^{k} \sum_{j=0}^{k} b_{ij} x_1^i x_2^j$$

$$p(x) = \sum_{i=0}^{k} b_i x^i$$

If weights are used in the least-squares regression, certain data points can be given a greater importance in the fitting process. This is used to give points closer to the phase border a greater weight in the fit. Making the residual smaller close to the border allows for a smoother transition between the different phases.

3.1 Regions

The regions are referred to as liquid, vapor and two-phase region. The liquid and vapor regions are divided into sub- and super-critical areas. The region of a certain point is decided by its p and h values. Figure 2 shows the phase diagram in the p-h plane with all of the regions.

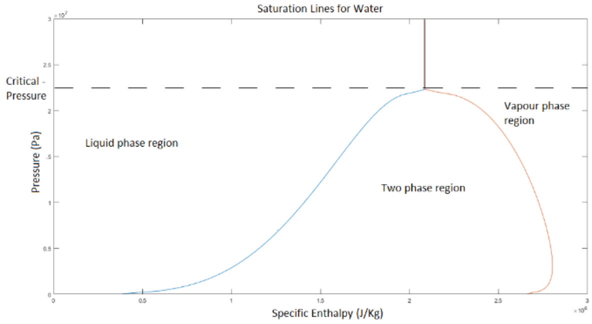

Figure 2 Phase diagram of water, saturation lines are drawn with approximated functions. Regions are divided into super- and sub-critical for both liquid and vapour.

Furthermore, it was noticed later in the process that accurate media calls were needed for very low pressures. Thus, a super low pressure region was added to the functions.

3.2 The smooth step function

The method used for making a smooth transition between regions is via a smooth step function S. The idea is to multiply the polynomial defining the function over a certain region with a function so that the function assumes the polynomial fit's value within the region and goes to zero outside this specific region. The desired properties of S are

$$\mathbf{S(x)} = \begin{cases} \mathbf{0}, & \mathbf{x \leq 0} \\ \mathbf{x}, & \mathbf{0 \leq x \leq 1} \\ \mathbf{1}, & \mathbf{1 \leq x} \end{cases}$$

The right- and left borders of the step has been chosen to 1 and 0 for easy implementation and the scaling can be done when calling the function by scaling the input parameter.

Since the overall goal with this implementation is to make the media implementation twice continuously differentiable the step function must also be so.

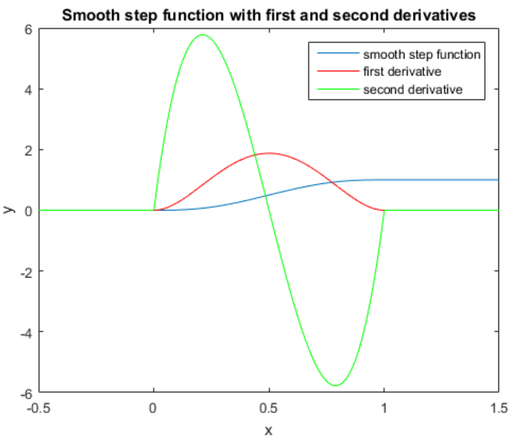

Figure 3 Smooth step function with its first and second derivative.

For this purpose, a generic 5th order polynomial can be used. If the boundary conditions on the derivatives and the function are applied the following solution is found.

$$\mathbf{S(x)} = \begin{cases} \mathbf{0}, & \mathbf{x \leq 0} \\ \mathbf{6x^5 - 15x^4 + 10x^3}, & \mathbf{0 \leq x \leq 1} \\ \mathbf{1}, & \mathbf{1 \leq x} \end{cases}$$

3.3 The approximating polynomials

The data needed for making the polynomial fits was extracted from Modelica.Media.WaterIF97.

The grid in the p-h plane that was used for data extraction was 100x100 points and linear along the h-axis with range [1.0e5, 4.0e6] (J/kg). A logarithmic scale was used for the p-axis with approximately the range [7.6e4, 3.0e7] (Pa).

For 2D-functions that use d, T and s as inputs, the response data from IF97 for constructing the functions that calculates these properties from p and h were used instead. This was done since it is hard to construct a grid that does not contain points outside of the domain of definition for these properties.

It is of extra importance that the polynomial fits have high accuracy close to the borders to other regions, since they are to be connected to another polynomial function there. Big differences in the values of the different surfaces close to the border will lead to a "leap" in the function value at the border. Even though the step function smoothes this leap out and makes sure the function is continuous it is of extra importance that this difference is made as small as possible since the model is to be used in optimization algorithms which can get stuck at inconsistencies like this. The approach used for handling this problem is to give data points at the borders between different regions a higher weight to make the linear regression generate polynomials which are accurate at the border. However this might cause "overshoots" in the rest of the region if the border points are weighted too much. This method was therefore used only when this phenomenon did not cause relatively large residuals inside the considered region.

Furthermore, weighting is used to make the least-square algorithm minimize the relative errors instead of absolute. Since some of the approximated functions range largely in value, data points which have small response reference values (close to zero) will get very large relative errors if weighting is not performed.

3.4 Accuracy of implemented media functions

Since 18 functions have been implemented, only a few important examples will be accounted for in this section.

3.4.1 Temperature

The temperature function is shown in Figure 4 and has a maximum relative error of around 0.8% as seen in Figure 5. The red line in the figure represents the phase border. The relative error is calculated as the percentage difference between the implemented approximation and IAPWS IF97.

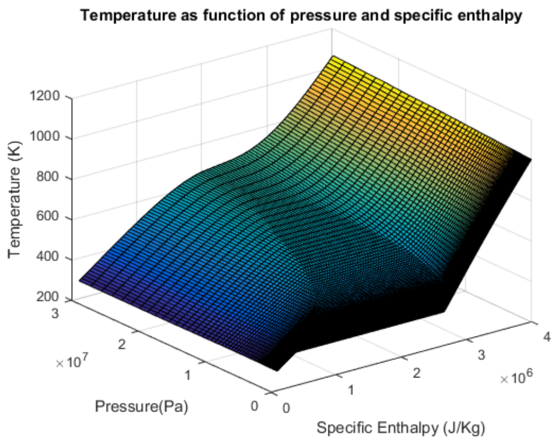

Figure 4 The approximated temperature function.

DOI 10.3384/ecp17132449

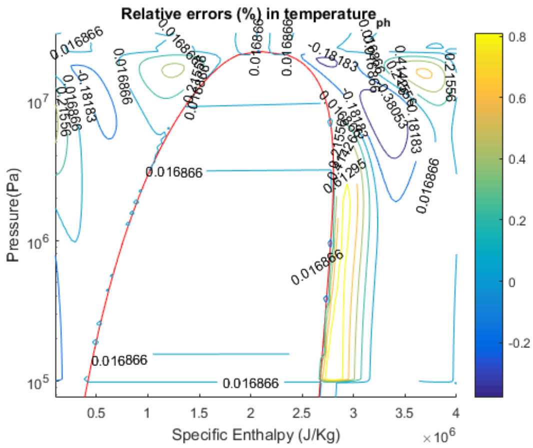

Figure 5 Contour plot of relative errors of the approximated temperature function.

There are a couple of interesting things to note from the relative error plot. At low pressures and high specific enthalpies there is a distinct drop in the relative error. This is because a new region was added for sub-1 bar pressures in the vapor region to get higher accuracy at components such as condensers which operate at very low pressures. Another thing to note is that the highest relative error is located after the phase border at the vapor side. The coefficients here have been weighted in a way to be consistent with the saturated properties at the phase border. This weighting might cause this bulge as the least squares-algorithm prioritizes minimizing the error at the border instead of inside the region but with the benefit of better consistency at the phase border.

3.4.2 Density

The density function and the corresponding relative errors can be viewed in Figure 6 and 7, respectively. As can be seen, there is a "spike" in the relative error around the critical point, which is due the connection of the three regions, and the value there is an interpolation between the function approximations in all three regions.

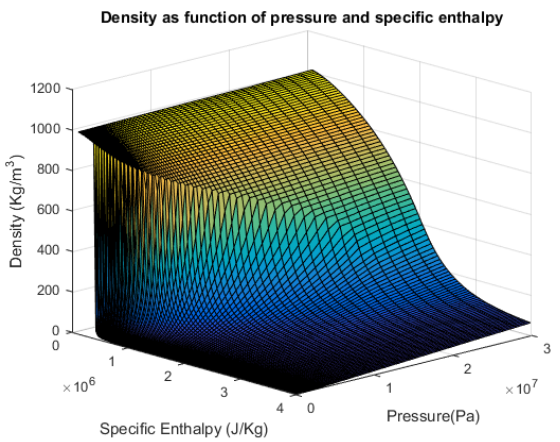

Figure 6 The approximated density function.

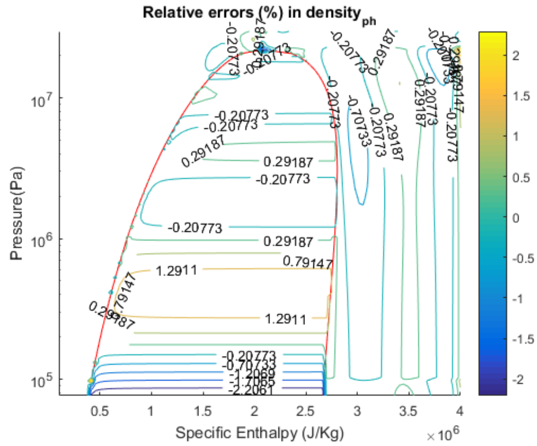

Figure 7 Contour plot of relative errors of the approximated density function

The same behavior at low pressures in the vapor region can be seen in the relative error plot as in the temperature function due to the same reasons, that is, an added region at low pressures. The spike in relative errors is due to the fact that 3 regions meet at the critical point. If the density function is compared with the temperature function, which has a similar point, it can be seen that the temperature function is rather flat at the critical point where the density is rather steep.

4 Optimization benchmarking case: Start-up of a Heat Recovery Steam Generator (HRSG)

For testing the implementation in optimization applications, a model describing a start-up phase of a heat recovery steam generator (HRSG) has been chosen.

4.1 Description of the HRSG model

The model used in this thesis has been built upon a model developed for a tutorial, for further information and material from this tutorial please see (Larsson, 2015). A similar model and optimization problem is investigated thoroughly in a master thesis previously written in cooperation with Modelon (Runvik, 2014).

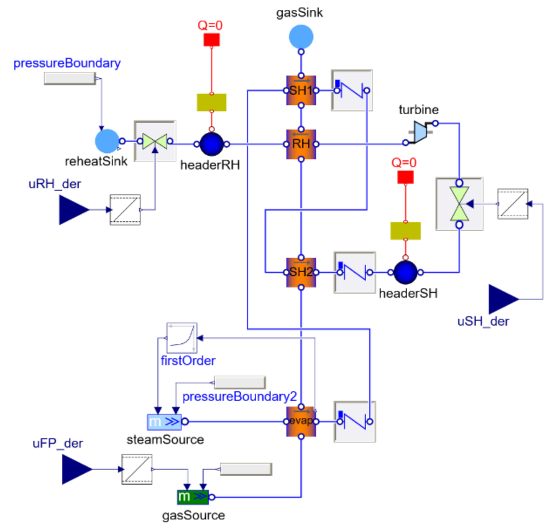

Figure 8 Model diagram view of the system used in the optimization.

The main working components of the model are a series of heat exchangers transferring heat from a flue gas source to the water medium until the steam reaches a desired working state. The flue gas is led into an evaporator where the water is evaporated into steam. A feedback loop connected to the evaporator keeps the water level in the evaporator on a constant level. From here, the steam goes through two superheaters where the steam pressure and temperature increase to reach the desired working levels. After superheater 2 the steam is collected in a superheater header, where in reality the pipes are collected and the steam can be redirected into a turbine step. There is also a wall model connected to the header. One of the main issues with the start-up phase of the power plant is the exposure of thermal stresses in the components, and thus this has to be modelled. The wall models allows for the modelling of these stresses. There is a valve located after the header which can be used to control the temperature and pressure inside the header. When the steam has reached high enough temperature and pressure it can instead of going through the control valve, be redirected into a turbine step. To maximize energy output of the plant, the steam is thereafter led into a reheater step. After going through the reheater header it could once again be led through another turbine step. Again, a control valve is added to be able to control the pressure and the temperature inside the header.

The controllable inputs of the model are the firing power of the gas source and the opening of the valves located after superheater 2 and the reheater header. These inputs can be used to control the pressure and temperature inside the headers and consequently can be used to limit the thermal stresses inside the header walls. As can be seen in Figure 8 integrator steps are added to the control inputs. This was done to be able to put constraints on the rate of change of the optimizing input signals.

4.2 Optimization problem formulation

The aim of the start-up is to take the plant from the initial operating point to another operating point as fast as possible without violating the problem constraints. The preferred properties of this process to reach this point are:

- Control the system from the initial operating point to a point where the steam in the plant has high enough quality to be redirected to turbines.
- The thermal stresses inside the header walls should be limited to extend the lifespan of the components.
- The controllable inputs have rate of change constraints which must be obeyed.

The optimal control problem defined over the time interval $[0, t_f]$ is stated as

$$\begin{aligned}\min \int_0^{t_f} & \mathrm{w_{TSH2}}(\mathrm{T_{SH2}} - \mathrm{T_{SH2ref}})^2 \\ & + \mathrm{w_{pSH2}}(\mathrm{p_{SH2}} - \mathrm{p_{SH2ref}})^2 \\ & + \mathrm{w_{pRH}}(\mathrm{p_{RH}} - \mathrm{p_{RHref}})^2 \\ & + \mathrm{w_{SH2v}}\dot{u}_{SH2v}^2 + \mathrm{w_{RHv}}\dot{u}_{RHv}^2 \\ & + \mathrm{w_b}\dot{u}_b^2 \,\mathrm{dt}\end{aligned}$$

subject to

$$\begin{gathered} model\ equations \\ dT_{SH2} < dT_{SH2}^{max} \\ dT_{RH} < dT_{RH}^{max} \\ |\dot{u}_{SH2v}| < \dot{u}_{SH2v}^{max} \\ |\dot{u}_{RHv}| < \dot{u}_{RHv}^{max} \\ |\dot{u}_b| < \dot{u}_b^{max} \end{gathered}$$

The first three terms of the objective function correspond to the penalties on temperature and pressure deviations from the desired values inside the heat exchangers (same as in the headers). $\mathrm{T_{SH2}}$ and $\mathrm{p_{SH2}}$ are the temperature and pressure inside superheater 2 ($\mathrm{T_{SH2ref}}$ and $\mathrm{p_{SH2ref}}$ the desired values), $\mathrm{p_{RH}}$ the pressure inside the reheater. w are the corresponding weights. The last three terms represent the derivatives of the control inputs, $\dot{u}_{RHv}$ is the reheater valve signal, $\dot{u}_{SH2v}$ the superheater 2 signal and $\dot{u}_b$ the boiler control signal.

The model equations are the equations that describe the dynamics of the system. dT_{SH2} is the thermal gradient in the superheater 2 header wall and dT_{RH} is its counterpart in the reheater header wall. The last three

DOI 10.3384/ecp17132449

constraints put upper and lower limits on the derivatives of the three control signals.

4.3 Optimization

A direct collocation method (Magnusson & Åkesson, 2015) is used for solving the optimization problem. The time horizon is divided into 12 elements, using 4 collocation points in each element. The element grid points are located so that they are closer together in the first part of the time horizon, to better capture the transient behavior at the beginning of the start-up. 3/4 of the elements are in the first 3/8 of the time horizon and 1/4 in the last 5/8.

Optimization statistics are summarized in Table 1. The optimization model has 8 continuous time states and 85 algebraic variables. This model is translated into a non-linear program with 5184 variables.

Table 1 Optimization statistics

DAE model	
Number of states	8
Number of algebraic variables	85
NLP model	
Total number of variables	5184
Solution statistics	
CPU-time in IPOPT (s)	1.45
CPU-time in NLP function evaluations (s)	1.56
Solution time (s)	3.11

4.4 Verification through simulation

To verify the result the optimized signals were extracted and used in a simulation experiment using Water-IF97 media functions. The trajectories for these simulations are displayed in Figures 9 and 10 alongside the trajectory from the optimization.

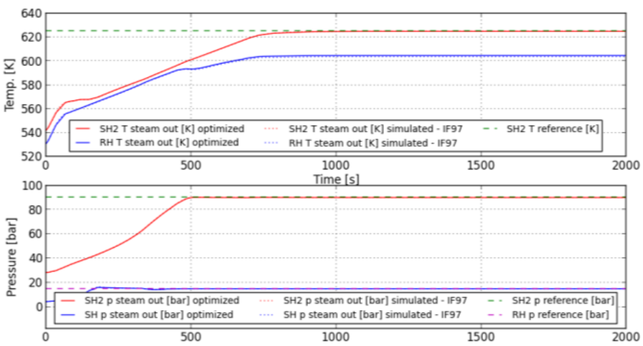

Figure 9 Temperature and pressure signals from optimization (solid) and simulation (dotted). In simulation, the optimal input signals are used as input, and the medium is modeled with Water IF97 thermodynamic property functions.

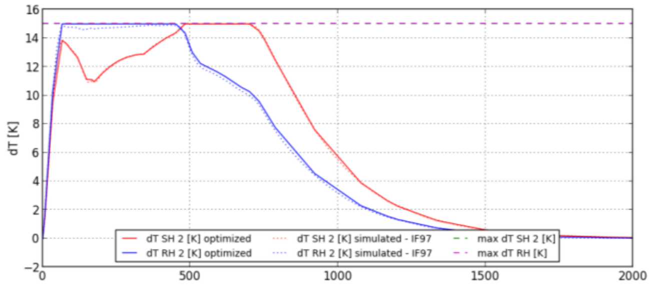

Figure 10 Metal wall temperature gradient signals from optimization (solid) and simulation (dotted). The dashed line represents the maximal allowed wall stress.

The simulation results match the optimized trajectory well, which indicates two things. Firstly, it indicates that the time discretization of the optimization model is sufficient to capture the dynamics of the model. Secondly, it indicates that the implemented media functions give very similar results to the IF97 functions.

5 Dynamic simulation

To verify that the media model can also handle industrial relevant dynamic simulation use cases, it was tested with large dynamic simulation examples in the Thermal Power Library. These tests expose the media implementation over various properties and under different operating conditions. As throughout this article, the Water-IF97 media implementation will be used as the reference medium.

Three use cases were set up:

1. Coal fired 400 MW electrical super-critical steam cycle that operates at a maximum pressure of 300 bar and 580 °C. The model consists of 5683 equations and 193 continuous time states.
2. Heat recovery steam generator (HRSG) that operates at a pressure around 84 bar and in a temperature interval of 175-500 °C. The model consists of 1616 equations and 39 continuous time states. In comparison with the optimization test case, this model includes more dynamics and describes the considered system more thoroughly.
3. Nuclear steam generator system that operates at a maximum pressure of 70bar and 285 °C. The model consists of 6708 equations and 147 continuous time states.

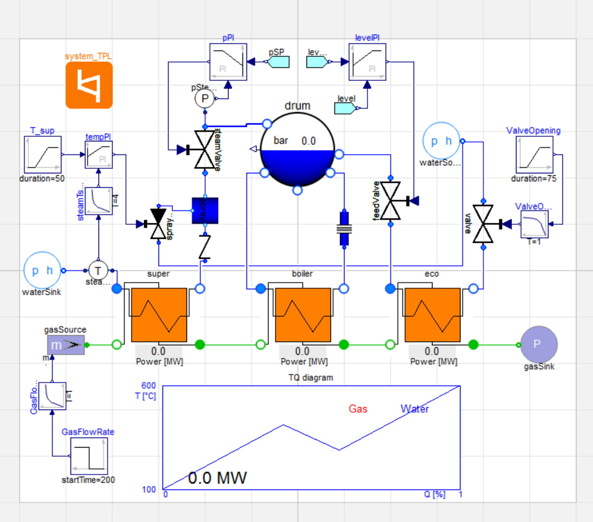

Figure 11 Model diagram view over the HRSG-model (test case 2)

The test cases were simulated on a standard laptop (Dell Latitude E7470, Intel i7-6600U) using the Modelica simulation tool Dymola 2017 with the solver Dassl and a tolerance of 1e-5.

5.1 Result

The result is summarized in the tables below. Using the new media implementation, a speed-up of up to 40% can be achieved in an industrial relevant large-scale power plant simulation. The difference in result of selected important variables is below 0.6% in use case 1 and 2 and 2.9% in use case 3, however it may be larger for certain intermediate pressure variables. The larger deviation in use case 3 is mainly due to a deviation in the isentropic efficiency calculation at the last turbine stage. This may be improved by dividing the specific entropy polynomial into regions at lower pressure in a similar way as was done with the density function. If trajectories from the simulations are compared, the results seem to match well as can be seen in Figure 11, showing the total power transferred from the exhaust gas to the steam through all three heat exchanger stages in use case 2.

The CPU-time is a combination of the computational effort that is required to do one integrator step and the number of steps. Even if a media implementation is faster the CPU-time of a simulation may increase due to an increase of the number of integrator steps. This may happen if the implementation contains variations due to the use of e.g. higher order polynomials or transitions between computational regions. The F-evaluations describe the number of function evaluations of all system equations. They are used in the integration process to evaluate derivatives and calculate numerical system Jacobians. Dassl use the Jacobian in its internal solver process (Petzold, 1982).

Table 2 Simulation statistics use case 1 (super-critical power plant simulated 25000s).

	Optimization media	WaterIF97 (reference)
Simulation statistics		
CPU-time (s):	24.5	34.6
Solver steps	825	862
F-evaluations	7310	8857
Jacobian-evaluations	125	158
Steady-state results		
Generated power	405.8 MW	408.2 MW
Condenser temperature	297.64K	297.60

Table 3 Simulation statistics use case 2 (HRSG simulated 200s).

	Optimization media	WaterIF97 (reference)
Simulation statistics		
CPU-time (s):	7.98	8.38
Solver steps	320	296
F-evaluations	3819	3412
Jacobian-evaluations	105	94
Steady-state results		
Total heat transfer	286.8 MW	287.1 MW
Steam outlet flow	10.69 kg/s	10.75 kg/s
Gas exhaust temperature	554.3K	554K

Table 4 Simulation statistics use case 3 (nuclear plant simulated 25000s).

	Optimization media	WaterIF97 (reference)
Simulation statistics		
CPU-time (s):	12.9	18.3
Solver steps	1112	1044
F-evaluations	14150	16048
Jacobian-evaluations	448	392
Steady-state results		
Generated power	432 MW	420 MW
Condenser temperature	33.67 °C	33.37 °C

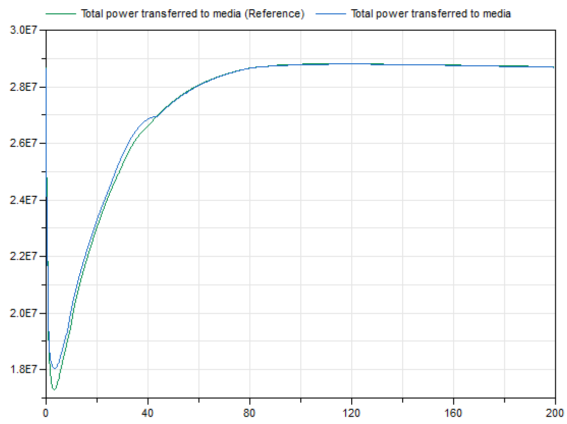

Figure 12 Total power transfer from exhaust gas to steam in use case 3.

6 Conclusions

Efficient optimization-friendly properties of water and steam covering sub-critical and super-critical regions have been implemented in Modelon's Modelica Thermal Power library 1.13. The new medium can bridge the gap between simulation and optimization and was tested against industrial relevant thermo-fluid systems. It was shown in the optimization benchmarking case that the implemented media functions could be used to provide results that coincide well with IF97 simulation results using the resulting optimal control inputs. This shows that the implementation suggested in this article can yield reliable results.

The simulation benchmarking test cases aimed at comparing the accuracy and performance of the new implementation with the existing Water-IF97 media implementation. The results from these simulations show that there are some slight deviations in the results between the implementations. However, the dynamics of the system are captured accurately and the relative errors are small. The largest deviations are observed at rapid transients. That there are deviations is expected as the implementation approximates the Water-IF97 standard. The question is whether these differences are small enough to yield acceptable results and in the tested simulation models this seems to be the case for a majority of the use cases. Comparing the simulation statistics of the large plant use cases shows that the new implementation is up to 40% faster.

6.1 Future work and possible improvements

It is desirable that the media is accurate at the phase borders. The length of the smoothing interval impacts the derivatives of the functions in the implementation. A smaller delta makes the transition between the polynomials go faster and hence making the function "less smooth" even though the implementation in theoretical sense still is twice continuously differentiable. However, making this parameter too big will instead decrease the accuracy in a larger region around the region borders.

When modelling thermodynamic properties, there are many natural laws to consider, which might not totally be satisfied by the approximations made as there is no check on whether such relations are fulfilled. An example of this is that by nature the density must increase with increased pressure if the temperature is kept constant. Iterative solvers that use gradients based on the function approximations might be affected if there are inconsistencies in such relations.

Furthermore, the choice of functional form in the least squares approximations might be investigated. There might be better forms of functions to represent the functions. In (Aute & Radermacher, 2014) the use of Chebyshev Rational polynomials is proposed for fast evaluation of thermodynamic properties. The use of different functional forms might be a way of making the implementation faster and more accurate.

For easy implementation of similar models describing the thermodynamic properties of other media than water, it is desirable to standardize the implementation. Ideally the whole work-flow would be automated so that the only thing that would have to be provided to create a new media model is the tables containing the thermodynamic property data. This has however been hard to achieve, as the many different functions that have been approximated have different shapes and appearances making it hard to construct an automated form for all these functions. It has been necessary to make specialized forms and adaptations for many of the functions to achieve good accuracy.

7 Acknowledgements

Fredrik Magnusson acknowledges support from the LCCC Linnaeus Center and eLLIIT Excellence Center at Lund University.

8 References

Andersson, J. (2013). *A general-purpose software framework for dynamic optimization.* Ph. D. thesis. Faculty of Engineering, KU Leuven, Leuven, Belgium.

Aute, V., & Radermacher, R. (2014). Standardized polynomials for fast evaluation of refrigerant thermophyiscal properties. *International Refrigeration and Air Conditioning Conference at Purdue.* Purdue, Indiana.

Casella, F., Donida, F., & Åkesson, J. (2011). Object-Oriented Modeling and Optimal Control: A Case Study in Power Plant Start-Up. *18th IFAC World Congress.*

Griewank, A., & Walther, A. (2008). *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation, second edition.* ISBN: 978-0-89871-659-7.

International Association for the Properties of Water and Steam. (2015). *Guideline on the Fast Calculation of Steam and Water Properties with the Spline-Based Table Look-Up Method (SBTL)* .

Kretzschmar, H.-J., & Wagner, W. (2008). *International Steam Tables. [electronic resource] : Properties of Water and Steam Based on the Industrial Formulation IAPWS-IF97.* Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg.

Kumar, S., & Mathur, T. (2014). Dynamic Load Scheduling of Optimization of Power Plants. *Advanced Control of INdustrial Processes (AdCONIP).* Hiroshima University, Hiroshima, Japan.

Larsson, P.-O. (2015, October). Report from the modelon tutorial at the 2015 modelica conference. Retrieved from http://www.modelon.com/blog/articles/report-from-the-modelon-tutorial-at-the-2015-modelica-conference/

Magnusson, F. (2016). *Numerical and symboolic methods for dynamic optimization.* PhD thesis, Lund University, Department of Automatic Control, Lund, Sweden.

Magnusson, F., & Åkesson, J. (2015). Dynamic Optimization in JModelica.org. *Processes, 3*(2), 471-496.

Nocedel, J., & Wright, S. (2006). *Numerical Optimization.* New York, NY: Springer New York.

Parini, P. (2015). *Object Oriented Modeling and Dynamic optimization of energy systems with application to combined-cycle power plant start-up.* Msc thesis, Politechnico di Milano, Milano.

Petzold, L. R. (1982). *A Description of DASSL: A Differential Algebraic System Solver.* Presented at IMACS World Congress, Montreal, Canada, August 8-1 3, 1982.

Runvik, H. (2014). *Modelling and start-up optimization of a coal-fired power plant.* Master's thesis, Lund University, Department of Automatic Control, Lund.

Schulze, C. (2014). *A Contribution to Numerically Efficient Modeling of Thermodynamic Systems.* PhD thesis, Technische Universität Braunschweig, Fakultät für Maschinenbau.

Thorade, M., & Saadat, A. (2013). *Partial derivatives of thermodynamic state properties for dynamic.* Environmental Earth Sciences, 70, 8, 3497-3503.

Wagner, W., Cooper, J., Dittmann, A., Kijima, J., Kretzschmar, H.-J., Kruse, A., . . . Trübenbach, J. (2000). The IAPWS Industrial Formulation 1997 for the Thermodynamic propertiies of Water and Steam. *J. Eng. Gas Turbines Power 122*, 150-182.

Velut, S., Larsson, P.-O., Runvik, H., Funqvist, J., Bohlin, M., Nilsson, A., & Modarrez Razavi, S. (2014). Production Planning for Distributed District Heating Networks. *11th International Modelica 2015 Conference.* Versailles, France.

Windahl, J., Prölss, K., Bosmans, M., Tummescheit, H., van Es, E., & Sewgobind, A. (2014). MultiComponentMultiPhase - A fraework for thermodynamics in Modelica. *Proceedings of the 11th International Modelica Conference.* Versailles, France.

Zimmer, D., Otter, M., Elmqvist, H., & Kurzbach, G. (2014). Custom Annotations: Handling Meta-Information in Modelica. *Proceedings of the 10th International Modelica 2014 Conference.* Lund, Sweden.

Åberg, M. (2016). *Optimisation-friendly modelling of thermodynamic properties of media.* Master's thesis, Lund University, Department of Automatic Control, Lund. Retrieved from http://lup.lub.lu.se/student-papers/record/8888181

Åkesson, J. (2008). Optimica - An Extension of Modelica Supporting Dynamic Optimization. *Proceedings of the 8th International Modelica 2008 Conference.* Bielefeld, Germany.

DOI
10.3384/ecp17132449