P R E P R I N T – ICPWS XV
Berlin, September 8–11, 2008

Fast Calculation of Thermodynamic Properties of Water and Steam in Process Modelling using Spline Interpolation

Matthias Kunick[a], Hans-Joachim Kretzschmar[a], and
Uwe Gampe[b]

[a] Department of Technical Thermodynamics,
Zittau/Goerlitz University of Applied Sciences, 02763 Zittau, Germany
[b] Institute for Power Engineering, Chair of Thermal Power Machinery and Plants,
Technical University of Dresden, 01062 Dresden, Germany

Email: matthiaskunick@hotmail.com, hj.kretzschmar@hs-zigr.de, uwe.gampe@tu-dresden.de

The designing and optimising of advanced power cycles and processes requires fast and accurate methods for calculating thermodynamic properties. Precise fundamental equations for most working fluids have already been developed. In industrial software applications such as process optimisations with heat balance calculations, the calculation of these equations requires enormous computing time. In Computational Fluid Dynamics (CFD), ideal-gas equations or other simple algorithms are often used to achieve reasonable computing times, but this procedure leads to inaccuracies in the process calculation. IAPWS therefore established a task group in 2007 to develop extremely fast and accurate property algorithms for water and steam. This paper gives an overview of the aims, initial results achieved at the Zittau/Goerlitz University of Applied Sciences and of future tasks. An advanced method for the calculation of thermodynamic properties using spline interpolation is currently in development. An initial study has been done with the calculation of $T(p,h)$ and $h(p,T)$ in the steam region 2 of IAPWS-IF97.

Introduction

Optimising heat cycles and calculating non-stationary processes require extremely fast algorithms for thermodynamic properties of working fluids because they are frequently used in the inner iteration cycles of the process calculations. The IAPWS-IF97 [1,2] contains very fast and accurate equations. For Computational Fluid Dynamics (CFD), however, even IAPWS-IF97 is too slow. Therefore in CFD, fluid properties are often calculated with simple equations, for example with the ideal gas equation. Depending on the range of state this procedure leads to inaccuracies in the process calculation. IAPWS therefore established a task group in 2007 for developing fast property algorithms for water and steam. In the past, table look-up methods have been developed to calculate fluid properties faster and with reasonable accuracy. One of these methods, the Tabular Taylor Series Expansion Method (TTSE) [3], was adopted by IAPWS as a guideline in 2003 [4]. One disadvantage of this method is that it does not represent the property surfaces in a continuous form.

In order to calculate fluid properties as fast as or faster as TTSE but with continuously represented property surfaces, two-dimensional splines can be used in a table look up method.

In the first step a bi-quadratic spline surface for the function $T_2^{\mathrm{SPL}}\left(p,h\right)$ for region 2 of IAPWS-IF97 was developed. Additionally, $h_2^{\mathrm{INV}}\left(p,T\right)$ was obtained by solving $T_2^{\mathrm{SPL}}\left(p,h\right)$ in terms of h.

This paper gives a brief introduction into spline-based fluid property calculations and discusses its characteristics in comparison to IAPWS-IF97 and TTSE.

The main focus of this work is the development of an extremely fast spline-based algorithm for calculating thermodynamic properties.

An investigation to determine spline-interpolation algorithms for thermodynamic properties not only for water and steam but also for other pure fluids and mixtures is currently being conducted.

Aims of the Project

In light of the requirements of modern calculation procedures such as solver-based heat cycle calculation and CFD, a method for fast and accurate calculation of fluid properties should be developed. The resulting algorithm shall substitute the calculation of properties from fundamental equations when high calculation speed and high accuracy is required. Because of iterative procedures in the superior process calculation, steadiness of the supplied functions and of their first derivatives is required.

Another point of interest is the numerical consistency between forward and backward functions such as $h(p, T)$ and $T(p, h)$. In complex calculations, such as simulations of transient processes, numerical consistency is important.

The algorithm should be usable for working fluid mixtures, as well.

These requirements lead to spline-based table look-up methods. In order to minimize computation time, fast search algorithms and data handling methods must be applied.

For each property function, the corresponding look-up table must be pre-processed. A software tool will be created which will generate these tables and necessary functions directly as usable source code or software libraries.

Spline Interpolation of Thermodynamic Properties

This section gives a brief description of how a spline-based calculation of fluid properties can be introduced into process calculations.

If a property z is determined as a function of the variables x_1 and x_2 then a grid of values of these properties can be created as shown in Fig. 1, where I and J denote the number of grid lines along x_1 and x_2, respectively. Consequently the grid consists of $I \times J$ nodes and $(I-1)\times(J-1)$ sub-rectangles.

It is advantageous if the grid lines are equidistant, as this enables us to determine the corresponding sub-rectangle in the grid from a given point (x_1, x_2) without a comprehensive search algorithm. This is important because it would slow down the computing time of $z^{\mathrm{SPL}}\left(x_1, x_2\right)$. A sub-division of the grid into several equidistant grids can be done as shown in Fig. 1. The density of the grid can then be changed locally in order to optimise the accuracy of the resulting spline function.

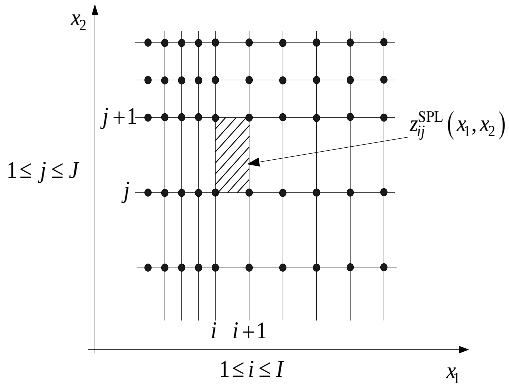

Figure 1: Grid of nodes

Spline functions $z_{ij}^{\mathrm{SPL}}\left(x_1, x_2\right)$ of the same structure must be defined for each sub-rectangle (i, j) (see Fig. 1). Depending on the requirements of the superior process calculation, a function must be chosen which is able to represent the fluid surface properly. A certain accuracy of $z_{ij}^{\mathrm{SPL}}\left(x_1, x_2\right)$ in comparison to $z^{\mathrm{EOS}}\left(x_1, x_2\right)$, the property function derived from the equation of state, is required first. A second demand comes from the solver-based calculation algorithms, such as Newton's method for non-linear systems, which are usually applied in heat cycle calculation programs. Normally the iterative procedures require continuous property functions and continuous first derivatives.

The simplest function which is able to satisfy these requirements is the bi-quadratic polynomial. It can be written as:

$$z_{ij}^{\mathrm{SPL}}\left(x_1, x_2\right) = \sum_{k=1}^{3}\sum_{l=1}^{3} a_{ijkl}\left(x_1 - x_{1i}\right)^{k-1}\left(x_2 - x_{2j}\right)^{l-1} \quad (1)$$

or

$$\begin{aligned} z_{ij}^{\mathrm{SPL}}\left(x_1, x_2\right) = {} & a_{ij11} + a_{ij21}\Delta x_{1i} + a_{31}\Delta x_{1i}^2 \\ & + a_{ij12}\Delta x_{2j} + a_{ij22}\Delta x_{1i}\Delta x_{2j} + a_{32}\Delta x_{1i}^2\Delta x_{2j} \quad (2) \\ & + a_{ij13}\Delta x_{2j}^2 + a_{ij23}\Delta x_{1i}\Delta x_{2j}^2 + a_{33}\Delta x_{1i}^2\Delta x_{2j}^2 \end{aligned}$$

with $\Delta x_{1i} = \left(x_1 - x_{1i}\right)$ and $\Delta x_{2j} = \left(x_2 - x_{2j}\right)$.

The coefficients for this function must be determined so that the resulting function z is continuously differentiable at least once.

To determine the $9\times(I-1)(J-1)$ coefficients, 9 for each sub-rectangle, the same number of conditions must be given. There are several possible ways of calculating the coefficients. The choice of the conditions influences accuracy and steadiness of the resulting spline function.

To create a spline function while preserving its optimal shape, it is recommended in [6] to have knots which are different from nodes. This means that a second grid of knots must be created. This should be done as shown in Figure 2.

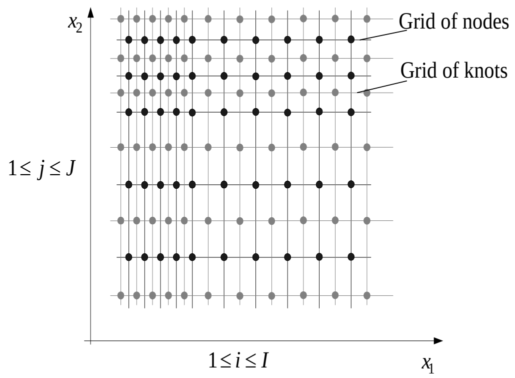

Figure 2: Grid of nodes and grid of knots

Fig. 3 illustrates the relations between the grids; here (′) denotes the grid of knots.

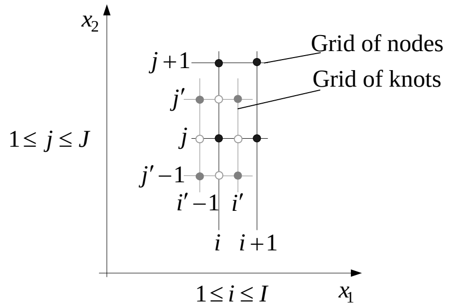

Figure 3: Sub-rectangle in grid of nodes and grid of knots

The interpolation requirement is to be fulfilled at (i, j). The derivatives $(\partial z / \partial x_1)_{x_2}$ at $(i'-1, j)$ and (i', j) as well as $(\partial z / \partial x_2)_{x_1}$ at $(i, j'-1)$ and (i, j'), marked with white circles, must be equal to the corresponding derivatives of the neighbouring sub-rectangles. In addition, the crossed derivatives $\left(\partial^2 z / (\partial x_1 \partial x_2)\right)$ at the four corner points $(i'-1, j'-1)$, $(i'-1, j')$, $(i', j'-1)$ and (i', j') must be equal to the corresponding derivatives of the neighbouring sub-rectangles. The system of equations can be solved for given derivatives at the boundary knots to obtain the coefficients of the spline polynomials.

The continuous behaviour at the nodes as well as at the boundaries between the sub-rectangles and their first derivatives can be mathematically proven [6].

Since all necessary function values $z_{i,j}(x_1, x_2)$ and derivatives $(\partial z / \partial x_1)_{x_2}$ and $(\partial z / \partial x_2)_{x_1}$ can be calculated directly from the fundamental equation, it is possible to create a spline with this approach.

An inverse function for $x_1(z, x_2)$ or $x_2(z, x_1)$ can be obtained by solving $z_{ij}^{\mathrm{SPL}}(x_1, x_2)$, Eq. (2), in terms of either x_1 or x_2.

For example $x_1(z, x_2)$ in a sub-rectangle (i, j) can be calculated from

$$x_{1,ij}^{\mathrm{INV}}(z, x_2) = \frac{\left(-B \pm \sqrt{B^2 - 4AC}\right)}{2A} + x_{1i} \qquad (3)$$

with

$$A = a_{ij31} + \Delta x_{2j}\left(a_{ij32} + a_{ij33}\Delta x_{2j}\right),$$

$$B = a_{ij21} + \Delta x_{2j}\left(a_{ij22} + a_{ij23}\Delta x_{2j}\right) \text{ and}$$

$$C = a_{ij11} + \Delta x_{2j}\left(a_{ij12} + a_{ij13}\Delta x_{2j}\right) - z\,,$$

where Δx_{2j} is given in Eq. (2).

The calculation of this equation consumes more computing time than the calculation of $z_{ij}^{\mathrm{SPL}}(x_1, x_2)$ since the operations square root and division are slower than multiplications. But the advantage is that $x_{1,ij}^{\mathrm{INV}}(z, x_2)$ is completely numerically consistent to the spline $z_{ij}^{\mathrm{SPL}}(x_1, x_2)$.

Spline-Function *T*(*p*, *h*) and Inverse Spline Function *h*(*p*, *T*) for IAPWS-IF97 Region 2

Especially in heat cycle calculations thermodynamic properties are frequently calculated from *p* and *h*. In order to find out about the quality of a spline-based calculation of thermodynamic properties with the approach described in the section above, a spline function $T_2^{\mathrm{SPL}}(p, h)$ for IAPWS-IF97 region 2 has been created. Furthermore a function $h_2^{\mathrm{INV}}(p, T)$ was prepared by solving the spline function $T_2^{\mathrm{SPL}}(p, h)$ in terms of *h*. Consequently $h_2^{\mathrm{INV}}(p, T)$ is completely numerically consistent to $T_2^{\mathrm{SPL}}(p, h)$.

In the first step the *p*-*h*-grid was given and not optimised for accuracy. To reach the required accuracy the grid has been created as shown in Fig. 3 above. The function $T_2^{\mathrm{IF97}}(p, h)$ has been iterated from the fundamental equation for all nodes in the range

0.000611 MPa $< p \le 100$ MPa and

$2500.9\ \mathrm{kJ\,kg^{-1}} \le h \le 4161\ \mathrm{kJ\,kg^{-1}}$.

Extrapolation was necessary because the spline algorithm currently in use is usable for rectangular grids only. The subdivision of the grid is also indicated in Fig. 3. Table 1 shows the number of grid lines in the corresponding ranges. The resulting grid consists of 60,000 nodes.

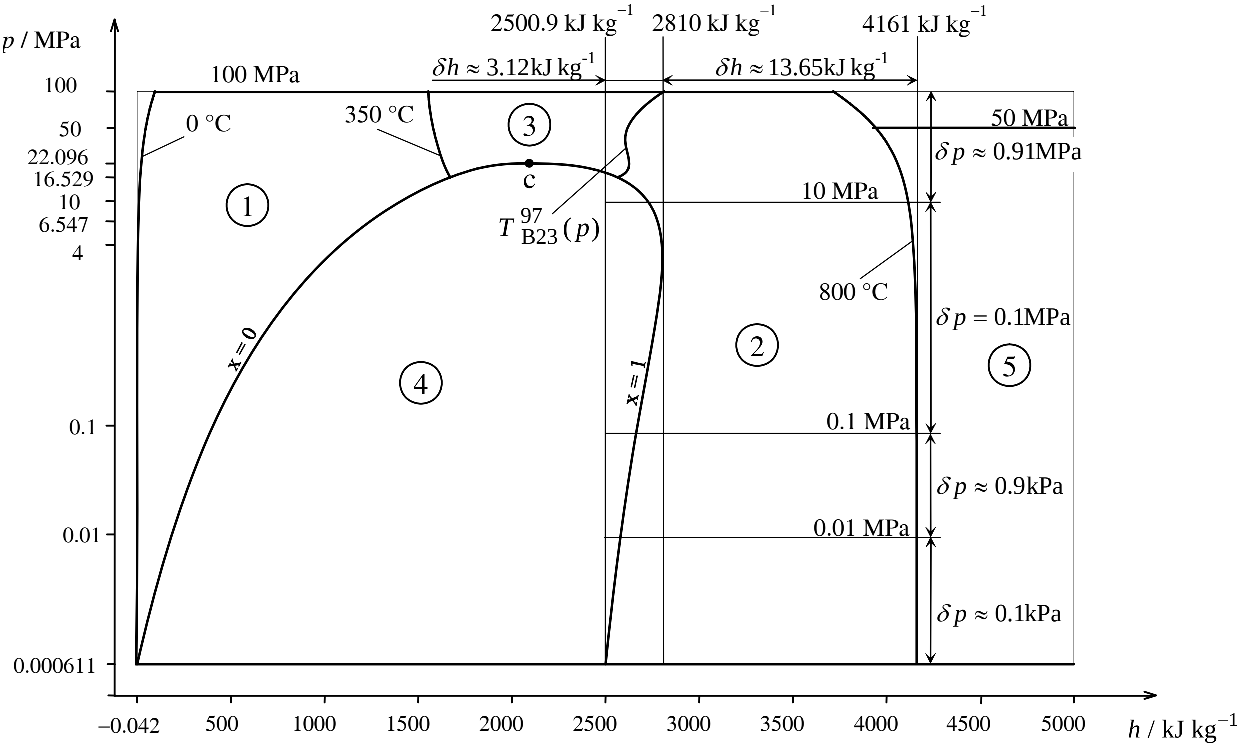

Figure 3: lg *p*-*h*-diagram with sub-divided *p*-*h* grid according to Table 1, where δp and δh indicate the distance along *p* and *h* , respectively

The calculation of the entire spline function $T_2^{\mathrm{SPL}}(p,h)$ gives a maximum relative deviation of 1×10^{-3}% in comparison with $T_2^{\mathrm{IF97}}(p,h)$ iterated from the fundamental equation of IAPWS-IF97 region 2. This means that the values of $T_2^{\mathrm{IF97}}(p,h)$ of IAPWS-IF97 are represented by 5 significant figures.

Table 1: Number of grid lines in *p*-*h*-grid for region 2

Range	*p*-grid lines
0.000611 MPa ≤ p ≤ 0.01 MPa	100
0.01 MPa ≤ p ≤ 0.1 MPa	100
0.1 MPa ≤ p ≤ 10 MPa	100
10 MPa ≤ p ≤ 100 MPa	100
Range	*h*-grid lines
2500.9 kJ kg^{-1} ≤ h ≤ 2810 kJ kg^{-1}	50
2810 kJ kg^{-1} ≤ h ≤ 4161 kJ kg^{-1}	100

Computing time comparisons

The computing time comparisons were carried out with a Pentium Xeon 3.2 GHz PC and Microsoft Windows XP operating system. Using the IAPWS software NIFBENCH [1], the computing speed of the developed spline-interpolation algorithms was compared to IAPWS-IF97 and to the TTSE method.

First, the function $T_2^{\mathrm{SPL}}(p,h)$ was compared to the corresponding IAPWS-IF97 backward equations $T_2^{\mathrm{97BW}}(p,h)$ and to the TTSE function $T^{\mathrm{TTSE}}(p,h)$ [5]. Table 2 shows the average computing times. They were determined for 100,000 state points arbitrarily distributed in IAPWS-IF97 region 2. As can be seen, the spline function is twice as fast as IAPWS-IF97 backward equations. The TTSE is even slower than IAPWS-IF97. The reason for this is the search algorithm of TTSE and its internal cell-finding logic using previous results. For given state points (p,h) close to each other, however, the TTSE function would be 2.1 times faster than IAPWS-IF97.

Table 2: Computing times for *T*(*p*, *h*) in µs

$T_2^{\mathrm{SPL}}(p,h)$	$T_2^{\mathrm{97BW}}(p,h)$	$T^{\mathrm{TTSE}}(p,h)$
0.056	0.114	0.178

An interesting result of the investigations is that the computing speed of the spline interpolation is nearly independent of the grid size and therefore of the number of nodes.

Furthermore the computing speed of the inverse spline function $h_2^{\mathrm{INV}}(p,T)$ was investigated. The computing times of $h_2^{\mathrm{IF97}}(p,T)$ and $h_2^{\mathrm{INV}}(p,T)$ are listed in Table 3. It can be seen that $h_2^{\mathrm{INV}}(p,T)$ is 1.2 times faster than $h_2^{\mathrm{IF97}}(p,T)$, but completely numerically consistent to $T_2^{\mathrm{SPL}}(p,h)$. The reason for this relatively low factor is that for evaluating $h_2^{\mathrm{INV}}(p,T)$ from $T_2^{\mathrm{SPL}}(p,h)$ an auxiliary spline function, $h_2^{\mathrm{SPL}}(p,T)$, is used for generating an initial estimate for *h*. Then, in the *p*-*h*-grid, the corresponding sub-rectangle is determined by solving $T_2^{\mathrm{SPL}}(p,h)$ in terms of *h* and comparing *h* to the neighbouring nodes until $h_{ij} \le h \le h_{ij+1}$ is fulfilled. The square root operation in Eq. (3) also slows down computation speed.

If greater computing speed for $h_2(p,T)$ is required and numerical consistency with $T_2^{\mathrm{SPL}}(p,h)$ does not need to be 100%, a separate spline function $h_2^{\mathrm{SPL}}(p,T)$ could be generated.

Table 3: Computing times for *h*(*p*, *T*) in µs

$h_2^{\mathrm{INV}}(p,T)$	$h_2^{\mathrm{IF97}}(p,T)$	$h^{\mathrm{TTSE}}(p,T)$
0.202	0.242	0.237

Summary and Outlook

The first results of this project show that spline functions can be used to represent thermodynamic properties. It has been proven that a reduction in computing time is possible while at the same time achieving high accuracy and complete numerical consistency. The computing speed is nearly independent of the size of the grid. This enables both high accuracy and low computing times.

A continuous spline function $T_2^{\mathrm{SPL}}(p,h)$ for region 2 has been developed. By solving $T_2^{\mathrm{SPL}}(p,h)$ in terms of *h*, the function $h_2^{\mathrm{INV}}(p,T)$, which is completely numerically consistent to $T_2^{\mathrm{SPL}}(p,h)$, could be obtained. The computational speed of these two functions is considerably faster than that of the IAPWS-IF97 fundamental or backward equations. The data grid of the spline-polynomials was created so that both functions represent IAPWS-IF97 with high accuracy. Due to special data handling and simple search algorithms, this method is even faster than the TTSE method.

Now the algorithm must be modified to enable the creation of spline functions from non-rectangular grids. An algorithm for grid optimisation is also necessary in order to reach the required accuracy and to reduce the amount of memory needed.

In order to make use of the described method in calculations of non-stationary processes, spline functions from *v*-*u* and *v*-*h* grids will be created.

The use of other spline functions, such as bi-cubic polynomials, is also intended for two-dimensional functions.

Spline algorithms are a very promising method for increasing the computing speed in calculating thermodynamic properties of mixtures.

To make spline-based table look-up methods available for industrial software applications, the necessary algorithms need to be provided in a convenient form. The software tool FluidGrid, currently in development, should meet this need. This tool will provide an interface which allows the user to prepare spline-based property libraries. This will be possible for available equations of state provided by the user in a dynamic link library. For given range of state and required accuracy FluidGrid will create and optimise a data grid and the corresponding spline function. Finally the software will provide automatically generated source code which can then be linked to and compiled in the user’s application.

Literature

[1] *IAPWS:*
Release on the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam, available at the IAPWS website http://www.iapws.org (2007).

[2] *Wagner, W.:*
The IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam, Journal of Engineering for Gas Turbines and Power Volume 122, 150-182 (2000).

[3] *Miyagawa, K. and Hill, P. G.:*
Rapid and Accurate Calculation of Water and Steam Properties Using the Tabular Taylor Series Expansion Method, Journal of Engineering for Gas Turbines and Power Volume 123, Issue 3, pp. 707-712 (2001).

[4] *IAPWS:*
Guideline on the Tabular Taylor Series Expansion (TTSE) Method for Calculation of Thermodynamic Properties of Water and Steam Applied to IAPWS-95 as an Example, available at the IAPWS website http://www.iapws.org (2003).

[5] *Miyagawa, K.:*
Software for the Tabular Taylor Series Expansion Method (TTSE), Personal communication (2008).

[6] *H. Späth:*
Two-Dimensional Spline Interpolation Algorithms, Oldenburg (1995).