

# User guide for Matlab class thermo

Are Mjaavatten, October 2020

Revised March 2023

The Matlab m-files mentioned in this note are all available from GitHub:

<https://github.com/are-mj/thermodynamics>.

Several authors have used methods developed at Ruhr-Universität Bochum (Setzmann & Wagner, 1989) to design high accuracy models for the thermodynamic properties of single-component fluids. These models calculate the molar Helmholtz free energy as a function of temperature  $T$  and molar volume  $v$ , in a form that allows explicit expressions for all partial derivatives of arbitrary order. All these models have largely the same structure and can be calculated using the same Matlab function (`helmholtz.m`), but with separate parameter sets for the different fluids.

## The thermo class and object

The purpose of the Matlab class thermo is to provide a flexible and efficient interface to the `helmholtz` function and to enable efficient methods for solving a range of thermodynamic problems. A Matlab class contains the code for creating one or more Matlab objects. A thermo object is created by calling the class with the name of an available fluid. Example:

```
>> th = thermo('H2O');
```

See the chapter on Parameter files near the end of the document to see which species are available.

A call to thermo creates a thermo object, reads the appropriate parameter file and stores the parameters as *properties* of the object. The object also contains *methods* for calculating thermodynamic variables. Note that for a single-component fluid, two independent thermodynamic variables will uniquely determine the *thermodynamic state* and thus all thermodynamic properties.

For a full list of available properties, type:

```
>> thermo.properties
```

For a list of functions available in the thermo object, type:

```
>> thermo.methods
```

## Thermodynamic methods of the thermo object

### Tvcalc

The basic method in the thermos object is `Tvcalc(T,v)`, where  $T$  is the temperature in K and  $v$  is the molar volume in  $\text{m}^3/\text{kmol}$ . This function calls the `helmholtz` function with temperature and molar volume as arguments. The Helmholtz function returns the Helmholtz free energy and its partial derivatives up to order two or three. (Two is the default). The calculations are explicit (i.e., no iterations are needed) and fast. `Tvcalc` stores these values as object

properties together with a number of calculated thermodynamic variables. All variables are in SI units, with the exception that kmol is used instead of mol. For a temperature of 700K (superheated steam) and a molar volume of  $1\text{m}^3/\text{kmol}$  we write:

```
>> th.Tvcalc(700,1);
```

All properties can be accessed as needed with no extra calculations:

```
>> th.p           % Pressure (Pa)
ans =
    5.4459e+06
>> th.c           % Speed of sound (m/s)
ans =
    622.7041
```

## Tpcalc

In many practical situations the temperature and pressure are known, not molar volume. The Tpcalc method uses Newton's method to find the molar volume  $v$  that gives the specified pressure. By default, it starts the search at the initial molar volume for ideal gas at the given T and p.

```
>> th.Tpcalc(373,101325) % Boiling point at one atmosphere
```

It is easy to calculate the density from the molar mass Mw and the molar volume

```
>> Rho = th.Mw/th.v %kg/m3
Rho =
    0.5979
```

This is the density of the vapour at the atmospheric pressure boiling point. To find the liquid density, we have to supply a realistic initial molar volume as a third input to Tpcalc. The density of water is about  $1000\text{ kg/m}^3$ , so:

```
>> rho0 = 1000;
>> v0 = th.Mw/rho0
v0 =
    0.0180
>> th.Tpcalc(373,101325,v0)
>> Rho_liq = th.Mw/th.v
Rho_liq =
    958.4569
```

## Vapour- liquid phase behaviour

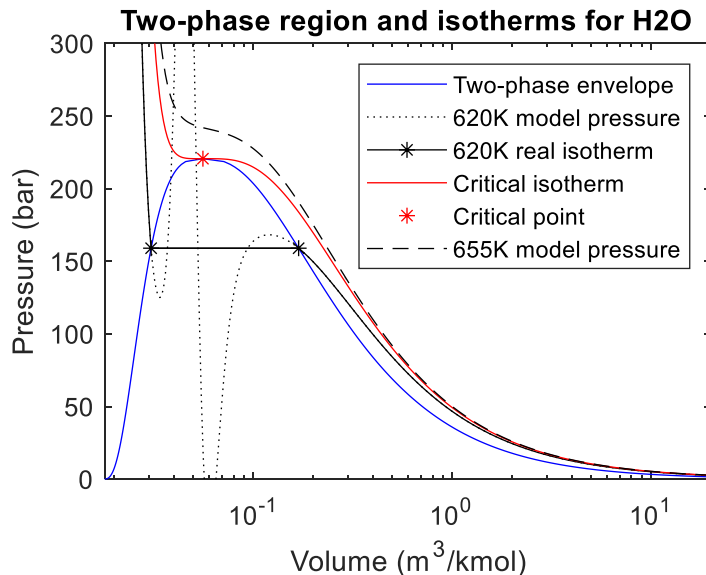
By keeping T constant and varying v we can use Tvcalc to plot lines of pressure at constant temperature, called isotherms. The figure below is created by the demo function

```
>> twophase_region('H2O')
```

It shows one isotherm passing through the two-phase region (black, full line), one in the superheated steam region (black, dashed) and one passing through the critical point (red). The envelope of the two-phase region (blue) is calculated using th.saturation.

The dotted, grey line shows the pressure calculated by Tvcalc inside the two-phase region. These values are artifacts of the model with no physical interpretation. The real situation is

that vapour and liquid coexist, with both phases having the same pressure. Each phase has a molar volume given by the black asterisk at the liquid and vapour end of the horizontal line. If the initial value  $v_0$  for `Tpcalc` is chosen somewhere between those values, `Tpcalc` may find one of the three unphysical solutions inside the two-phase region. If the user is aware of this problem, it is usually easy to avoid.



## phcalc and pscalc

These two methods find the temperature and molar volume that result in the given values for pressure and either molar enthalpy or entropy, respectively. They are less robust than `Tpcalc`, but will converge fast given good initial values. They use the current  $T$  and  $v$  values in the thermodynamic object as initial values.

These functions are useful if we work on processes where the pressure changes at constant enthalpy (e.g., pipes) or entropy (e.g., valves).

## Derivative order

By default, thermo calculates derivatives up to order two. Setting the property `max_order` to 3 forces the calculation of third order derivatives, at the cost of a longer calculation time.

Example:

```
>> th = thermo('H2O');
>> th.Tpcalc(400,1e5); th.f_TTT
ans =
     []
>> th.max_order=3;th.Tpcalc(400,1e5);th.f_TTT
ans =
    0.2000
>> T = linspace(400,500);v = linspace(10,40);
```

Execution time with max order 3:

```
>> tic;for i = 1:100;th.Tvcalc(T(i),v(i));end;toc
Elapsed time is 0.088023 seconds.
```

Execution time with max order 2:

```
>> th.max_order=2;
>> tic;for i = 1:100;th.Tvcalc(T(i),v(i));end;toc
Elapsed time is 0.024429 seconds
```

Third order derivatives are sometimes needed to calculate jacobians for solving problems involving second order derivatives.

## thermo object vs. separate function calls

Some user may find it conceptually easier to have separate functions for each thermodynamic variable, for instance:

```
h = enthalpy(T,p,'H2')
```

rather than

```
th2 = thermo('H2'); th2.Tpcalc(T,p)
h = th2.h
```

The advantage of the latter approach is that complex calculations need not be repeated if we need more than one property at a given  $(T, p)$  pair. Without extra function calls we can get, for example, the heat capacity at constant pressure and the speed of sound:

```
>> fprintf('Heat capacity: %.0f J/K/mol\n',th.cp)
Heat capacity: 28853 J/K/mol
>> fprintf('Speed of sound: %4.0f m/s\n',th.c)
Speed of sound: 1319 m/s
```

## Other thermodynamic functions

The derivatives of the helmholtz function available in the thermo object enables the efficient calculation of most thermodynamic processes. We have included a number of separate Matlab m-files demonstrating this flexibility:

### **choked\_flow.m**

State of choked gas flow, given upstream stagnation conditions. Assumes isentropic process and no condensation.

### **compressor.m**

Outlet conditions from gas compressor

### **twophase\_tank\_rhs.m**

Right-hand side for ODE system solving mass and energy balance and phase equilibrium for a two-phase tank being filled or emptied.

### **shocktube.m**

#### **shocktube\_ig.m**

See also: demo\_shocktube.m

Pressures and temperatures in idealised shock tube for real and ideal gas

### **shock\_from\_mach.m**

#### **shock\_from\_pressure.m**

#### **shock\_from\_velocity.m**

Calculate shock for real gases with various input variables

### **psat.m**

Vapour/liquid saturation pressure as a function of temperature. This can also be calculated by `th.saturation`, but `psat` can take T arrays as input, which is convenient for making plots.

## Demo files

### **demo\_gas\_tank\_system.m**

Simple simulation of filling a gas tank from another tank

### **twophase\_region.m**

Plot phase envelope and isotherms in a p-v plot

### **demo\_shocktube.m**

Calculate shock tube temperatures and pressures for a range of initial pressures

### **phase\_diagram.m**

Plots phase boundaries for the gas, liquid and solid regions in a T-p diagram. Currently available only for H<sub>2</sub>O and CO<sub>2</sub>.

## Parameter files

The files `parameter_xxx.m` contain parameters for species xxx.

### **parameters\_Air.m**

Air treated as a single species. Single-phase only. (Lemmon, Jacobsen, Penoncello, & Friend, 2000)

### **parameters\_Ar.m**

Argon. Single-phase only. (Tegeler, Span, & Wagner, 1999)

### **parameters\_CO2.m**

Carbon dioxide. Gas and liquid. (Span & Wagner, 1996) and (Giordano et al., 2006).

### **parameters\_H2O.m**

Water and steam. (Wagner & Pruss, 2002) and (Wagner et al., 2011)

### **parameters\_O2.m**

Oxygen. Single-phase only. (Lemmon, Jacobsen, Penoncello, & Friend, 2000)

### **parameters\_N2.m**

Nitrogen. Single-phase only. (Lemmon, Jacobsen, Penoncello, & Friend, 2000)

### **parameters\_H2.m**

#### **parameters\_paraH2.m**

#### **parameters\_orthoH2.m**

Hydrogen. Gas and liquid. (Leachman, Jacobsen, Penoncello, & Lemmon, 2009)

Parahydrogen and orthohydrogen are “spin isomers”. These are relevant for liquid and low temperature gas. The first file is for normal H<sub>2</sub>, which is the equilibrium combination at room temperature and higher: 75% orthohydrogen and 25% parahydrogen. See [Wikipedia](#) and (Leachman J. , 2015)

## Utility functions

`newton.m`

Solver for nonlinear equation systems using Newton's method.

## References

- Giordano, V. M., Datchi, F., & Dewaele, A. (2006). Melting curve and fluid equation of state of carbon dioxide at high pressure and high temperature. *The Journal of chemical physics*, 125(5), 054504. <https://arxiv.org/pdf/cond-mat/0607785.pdf>
- Leachman, J. (2015, July 15). *Why equilibrium hydrogen doesn't exist*. Retrieved from <https://hydrogen.wsu.edu/2015/06/22/why-equilibrium-hydrogen-doesnt-exist/>
- Leachman, J., Jacobsen, R., Penoncello, S., & Lemmon, E. (2009). Fundamental Equations of State for Parahydrogen, Normal Hydrogen, and ORTHOHYDROGEN. *Journal of Physical and Chemical Reference Data A*, 721-784. Retrieved from [https://www.researchgate.net/profile/Steve-Penoncello/publication/228339083\\_Fundamental\\_Equations\\_of\\_State\\_for\\_Parahydrogen\\_Normal\\_Hydrogen\\_and\\_Orthohydrogen/links/571a82e508ae7f552a47330e/Fundamental-Equations-of-State-for-Parahydrogen-Normal-Hydrogen-and-Orthohydrogen.pdf](https://www.researchgate.net/profile/Steve-Penoncello/publication/228339083_Fundamental_Equations_of_State_for_Parahydrogen_Normal_Hydrogen_and_Orthohydrogen/links/571a82e508ae7f552a47330e/Fundamental-Equations-of-State-for-Parahydrogen-Normal-Hydrogen-and-Orthohydrogen.pdf)
- Lemmon, E., Jacobsen, R., Penoncello, S., & Friend, D. (2000). Thermodynamic properties of air and mixtures of nitrogen, argon, and oxygen from 60 to 2000 K at pressures to 2000 MPa. *J. Phys. Chem. Ref. Data.*, 331-385. [http://thermophysics.ru/pdf\\_doc/refer331.pdf](http://thermophysics.ru/pdf_doc/refer331.pdf)
- Setzmann, U., & Wagner, W. (1989). A New Method for Optimizing the Structure of Thermodynamic Correlation Equations. *International Journal of Thermophysics*, 1103–1126. Retrieved from <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.471.8839&rep=rep1&type=pdf>
- Span, R., & Wagner, W. (1996). A New Equation of State for Carbon Dioxide ... *Journal of Physical and Chemical Reference Data*, 25, 1509-1596. Retrieved from <https://pdfs.semanticscholar.org/990f/7d99a7bfa2b851711a8f2add054728ee6922.pdf>
- Tegeler, C., Span, R., & Wagner, W. (1999). A new equation of state for argon covering the fluid region for temperatures from the melting line to 700 K at pressures up to 1000 MPa. *Journal of Physical and Chemical Reference Data*, 779-850. <https://www.nist.gov/system/files/documents/srd/jpcrd561.pdf>
- Wagner, W., & Pruss, A. (2002). The IAPWS Formulation 1995 for the Thermodynamic Properties of Ordinary Water. *Journal of Physical and Chemical Reference Data*, 387-535. Retrieved from <http://www.personal.psu.edu/mrh318/Wagner-Pruss-JPCRD-2002.pdf>
- Wagner, W. et al. (2011). New Equations for the Sublimation Pressure and Melting Pressure of H<sub>2</sub>O Ice Ih. *Journal of Physical and Chemical Reference Data* 40.4: 043103. [https://tsapps.nist.gov/publication/get\\_pdf.cfm?pub\\_id=909228](https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=909228)