NAME

iapws − library for light and heavy water properties according to IAPWS

LIBRARY

iapws library (*libiapws*, *−liapws*)

SYNOPSIS

Fortran **use iapws**

C **#include "iapws.h"**

Python **import pyiapws**

DESCRIPTION

Numerical implementation for reports:

- R2-83
  - [x] Tc in H2O and D2O
  - [x] pc in H2O and D2O
  - [x] rhoc in H2O and D2O
- G7-04
  - [x] kH
  - [x] kD
- R7-97
  - [x] Region 1
  - [] Region 2
  - [] Region 3
  - [x] Region 4
  - [] Region 5
- R11-24:
  - [x] Kw

NOTES

dp stands for double precision and it is an alias to real64 from the iso_fortran_env module.

Water phases:

l	liquid
v	vapor
c	super critical
s	saturation
n	unknown

References

[1] IAPWS, “Release on the Values of Temperature, Pressure and Density of Ordinary and Heavy Water Substances at Their Respective Critical Points.” IAPWS, St. Petersburg, Russia, 1992.

[2] “International Equations for the Saturation Properties of Ordinary Water Substance. Revised According to the International Temperature Scale of 1990. Addendum to J. Phys. Chem. Ref. Data 16, 893 (1987),”, Journal of Physical and Chemical Reference Data, vol. 22, no. 3, pp. 783–787, May 1993, doi: 10.1063/1.555926.

[3] A. H. Harvey and E. W. Lemmon, “Correlation for the Vapor Pressure of Heavy Water From the Triple Point to the Critical Point,” Journal of Physical and Chemical Reference Data, vol. 31, no. 1, pp. 173–181, Mar. 2002, doi: 10.1063/1.1430231.

[4] W. Wagner and A. Pruß, “The IAPWS Formulation 1995 for the Thermodynamic Properties of Ordinary Water Substance for General and Scientific Use,” Journal of Physical and Chemical Reference Data, vol. 31, no. 2, pp. 387–535, Jun. 2002, doi: 10.1063/1.1461829.

[5] R. Fernandez-Prini, J. L. Alvarez, and A. H. Harvey, “Henry’s Constants and Vapor–Liquid Distribution Constants for Gaseous Solutes in H2O and D2O at High Temperatures,” Journal of Physical Chemistry Reference Data, vol. 32, no. 2, pp. 903–916, 2003.

[6] IAPWS, “Guideline on the Henry’s Constant and Vapor-Liquid Distribution Constant for Gases in H2O and D2O at High Temperatures.” IAPWS, Kyoto, Japan, 2004.

[7] A. V. Bandura and S. N. Lvov, “The Ionization Constant of Water over Wide Ranges of Temperature and Density,” Journal of Physical and Chemical Reference Data, vol. 35, no. 1, pp. 15–30, Mar. 2006, doi: 10.1063/1.1928231.

[8] IAPWS, “Revised Release on the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam.” Lucerne, Switzerland, 2012.

[9] IAPWS, “Revised Release on the IAPWS Formulation 1995 for the Thermodynamic Properties of Ordinary Water Substance for General and Scientific Use.” Praque, Czech Republic, 2018.

[10] IAPWS, “Revised Release on the Ionization Constant of H2O.” Banff, Canada, 2019.

[11] H. Arcis et al., “Revised Parameters for the IAPWS Formulation for the Ionization Constant of Water Over a Wide Range of Temperatures and Densities, Including Near-Critical Conditions,” Journal of Physical and Chemical Reference Data, vol. 53, no. 2, p. 23103, May 2024, doi: 10.1063/5.0198792.

EXAMPLES

Example in Fortran:

```
program example_in_f
use iapws__common, only: dp, int32
use iapws
implicit none(type,external)

integer(int32) :: i, ngas
real(dp) :: T(1), kh_res(1), kd_res(1), wp_res(1), p(1)
real(dp) :: Ts(7), ps(7)
real(dp) :: x(3), y(3)
integer(int32) :: r(3)
character(len=1) :: s(3)
character(len=2) :: gas = "O2"
integer(int32) :: heavywater = 0
type(gas_type), pointer :: gases_list(:)
character(len=:), pointer :: gases_str

print *, '########################## IAPWS VERSION ##########################
print *, "version ", version()

print *, '########################## IAPWS R2-83 ##########################'
```

```
print "(a, f10.3, a)", "Tc in h2o=", Tc_H2O, " k"
print "(a, f10.3, a)", "pc in h2o=", pc_H2O, " mpa"
print "(a, f10.3, a)", "rhoc in h2o=", rhoc_H2O, " kg/m3"

print "(a, f10.3, a)", "Tc in D2O=", Tc_D2O, " k"
print "(a, f10.3, a)", "pc in D2O=", pc_D2O, " mpa"
print "(a, f10.3, a)", "rhoc in D2O=", rhoc_D2O, " kg/m3"
print *, ''

print *, '########################## IAPWS G7-04 ##########################'
! Compute kh and kd in H2O
T(1) = 25.0_dp + 273.15_dp
call kh(T, gas, heavywater, kh_res)
print "(A10, 1X, A10, 1X, A2, F10.1, A, 4X, A3, SP, F10.4)", "Gas=", gas, "T=

call kd(T, gas, heavywater, kd_res)
print "(A10, 1X, A10, 1X, A2, F10.1, A, 4X, A3, SP, F15.4)", "Gas=", gas, "T=

! Get and print available gases
heavywater = 0
ngas = ngases(heavywater)
gases_list => null()
gases_list => gases(heavywater)
gases_str => gases2(heavywater)
print *, "Gases in H2O: ", ngas
print *, gases_str
do i=1, ngas
print *, gases_list(i)%gas
enddo

heavywater = 1
ngas = ngases(heavywater)
gases_list => null()
gases_list => gases(heavywater)
gases_str => gases2(heavywater)
print *, "Gases in D2O: ", ngas
print *, gases_str
do i=1, ngas
print *, gases_list(i)%gas
enddo

print *, '########################## IAPWS R7-97 ##########################'
! Compute ps from Ts.
Ts(:) = [-1.0_dp, 25.0_dp, 100.0_dp, 200.0_dp, 300.0_dp, 360.0_dp, 374.0_dp]
Ts(:) = Ts(:) + 273.15_dp
call psat(Ts, ps)

do i=1, size(Ts)
print "(SP, F23.3, A3, 4X, F23.3, A3)", Ts(i), "K", ps(i), "MPa"
end do

! Compute Ts from ps
call Tsat(ps, Ts)
do i=1, size(Ts)
```

```
print "(SP, F23.3, A3, 4X, F23.3, A3)", Ts(i), "K", ps(i), "MPa"
end do

! Compute water properties at 280°C/8 Mpa
p(1) = 8.0_dp
T(1) = 273.15_dp + 280.0_dp
call wp(p, T, "v", wp_res)
print "(A5, F23.16, X, A)", "v(8MPa,280°C)=", wp_res(1)*1000.0_dp, "L/kg"

! Compute region and phase
x = [8.0_dp, 4.0_dp, 6.0_dp ]
y = [553.15_dp, 1200.0_dp, 2000.0_dp]
call wr(x, y, r)
call wph(x, y, s)
print *, r
print *, s

end program example_in_f
```

Example in C

```
#include <string.h>
#include <stdio.h>
#include "iapws.h"

int main(void){
double T = 25.0 + 273.15; /* in C*/
double p; /* p in Mpa */
char *gas = "O2";
double kh, kd, wp_res;
char **gases_list;
char *gases_str;
int ngas;
int i;
int heavywater = 0;
double x[3]= {8.0, 4.0, 6.0 };
double y[3] = {553.15, 1200.0, 2000.0};
int r[3];
char s[3];

printf("%s0, "########################## IAPWS VERSION ######################
printf("version %s0, iapws_version());

printf("%s0, "########################## IAPWS R2-83 ########################
printf("%s %10.3f %s0, "Tc in H2O", iapws_r283_Tc_H2O, "K");
printf("%s %10.3f %s0, "pc in H2O", iapws_r283_pc_H2O, "MPa");
printf("%s %10.3f %s0, "rhoc in H2O", iapws_r283_rhoc_H2O, "kg/m3");

printf("%s %10.3f %s0, "Tc in D2O", iapws_r283_Tc_D2O, "K");
printf("%s %10.3f %s0, "pc in D2O", iapws_r283_pc_D2O, "MPa");
printf("%s %10.3f %s0, "rhoc in D2O", iapws_r283_rhoc_D2O, "kg/m3");

printf("0);
```

```
printf("%s0, "########################## IAPWS G7-04 ##########################
/* Compute kh and kd in H2O*/
iapws_g704_kh(&T, gas, heavywater, &kh, strlen(gas), 1);
printf("Gas=%sT=%fKkh=%+10.4f0, gas, T, kh);

iapws_g704_kd(&T, gas, heavywater, &kd, strlen(gas), 1);
printf("Gas=%sT=%fKkd=%+15.4f0, gas, T, kd);

/* Get and print the available gases */
ngas = iapws_g704_ngases(heavywater);
gases_list = iapws_g704_gases(heavywater);
gases_str = iapws_g704_gases2(heavywater);
printf("Gases in H2O: %d0, ngas);
printf("%s0, gases_str);
for(i=0; i<ngas; i++){
    printf("%s0, gases_list[i]);
}

heavywater = 1;
ngas = iapws_g704_ngases(heavywater);
gases_list = iapws_g704_gases(heavywater);
gases_str = iapws_g704_gases2(heavywater);
printf("Gases in D2O: %d0, ngas);
printf("%s0, gases_str);
for(i=0; i<ngas; i++){
    printf("%s0, gases_list[i]);
}

printf("%s0, "########################## IAPWS R7-97 ##########################
double Ts[7] =  {-1.0, 25.0, 100.0, 200.0, 300.0, 360.0, 374.0};
double ps[7] = {1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0};
for(i=0; i<7; i++){
    Ts[i] = Ts[i] + 273.15;
}
iapws_r797_psat(7, Ts, ps);

for(i=0; i<7; i++){
    printf("%+23.3f %s %+23.3f %s0, Ts[i], "K", ps[i], "MPa");
}

iapws_r797_Tsat(7, ps, Ts);
for(i=0; i<7; i++){
    printf("%+23.3f %s %+23.3f %s0, Ts[i], "K", ps[i], "MPa");
}

T = 273.15 + 280.0;
p = 8.0;
iapws_r797_wp(&p, &T, "v", &wp_res, 1, 1);
printf("v(8MPa,280°C) = %+23.16f L/kg0, wp_res * 1000.0);

iapws_r797_wr(x, y, r, 3);
iapws_r797_wph(x, y, s, 3);
for(i=0; i<3; i++){
    printf("%i", r[i]);
```

```
}
printf("0);
for(i=0; i<3; i++){
    printf("%c", s[i]);
}
printf("0);

return 0;
}
```

Example in Python

```
r"""Example in python"""
import sys
sys.path.insert(0, "../py/src/")
import array
import numpy as np
import matplotlib.pyplot as plt
import pyiapws

print("########################## IAPWS VERSION ##########################")
print(pyiapws.__version__)

print("########################## IAPWS R2-83 ##########################")
print("Tc in H2O", pyiapws.Tc_H2O, "K")
print("pc in H2O", pyiapws.pc_H2O, "MPa")
print("rhoc in H2O", pyiapws.rhoc_H2O, "kg/m3")

print("Tc in D2O", pyiapws.Tc_D2O, "K")
print("pc in D2O", pyiapws.pc_D2O, "MPa")
print("rhoc in D2O", pyiapws.rhoc_D2O, "kg/m3")

print("")

print("########################## IAPWS G7-04 ##########################")
gas  = "O2"
T = array.array("d", (25.0+273.15,))

# Compute kh and kd in H2O
heavywater = False
k = pyiapws.kh(T, "O2", heavywater)
print(f"Gas={gas}T={T[0]}Kkh={k[0]:+10.4f}0)

k = pyiapws.kd(T, "O2", heavywater)
print(f"Gas={gas}T={T[0]}Kkd={k[0]:+10.4f}0)

# Get and print the available gases
heavywater = False
gases_list = pyiapws.gases(heavywater)
gases_str = pyiapws.gases2(heavywater)
ngas = pyiapws.ngases(heavywater)
print(f"Gases in H2O: {ngas:}")
print(gases_str)
for gas in gases_list:
    print(gas)
```

```
heavywater = True
gases_list = pyiapws.gases(heavywater)
gases_str = pyiapws.gases2(heavywater)
ngas = pyiapws.ngases(heavywater)
print(f"Gases in D2O: {ngas:}")
print(gases_str)
for gas in gases_list:
    print(gas)

style = {"marker":".", "ls":"", "ms":2}
T_KELVIN = 273.15
T = np.linspace(0.0, 360.0, 1000) + 273.15

solvent = {True: "D2O", False: "H2O"}

print("Generating plot for kh")
kname = "kh"
for HEAVYWATER in (False, True):
    print(solvent[HEAVYWATER])
    fig = plt.figure()
    ax = fig.add_subplot()
    ax.grid(visible=True, ls=':')
    ax.set_xlabel("T /°K")
    ax.set_ylabel("ln (kh/1GPa)")
    gases = pyiapws.gases(HEAVYWATER)
    for gas in gases:
        k = pyiapws.kh(T, gas, HEAVYWATER) / 1000.0
        ln_k = np.log(k)
        ax.plot(T, ln_k, label=gas, **style)
    ax.legend(ncol=3)
    fig.savefig(f"../media/g704-{kname:s}_{solvent[HEAVYWATER]}.png", dpi=100

print("Generating plot for kd")
kname = "kd"
for HEAVYWATER in (False, True):
    print(solvent[HEAVYWATER])
    fig = plt.figure()
    ax = fig.add_subplot()
    ax.grid(visible=True, ls=':')
    ax.set_xlabel("T /°K")
    ax.set_ylabel("ln kd")
    gases = pyiapws.gases(HEAVYWATER)
    for gas in gases:
        k = pyiapws.kd(T, gas, HEAVYWATER)
        ln_k = np.log(k)
        ax.plot(T, ln_k, label=gas, **style)
    ax.legend(ncol=3)
    fig.savefig(f"../media/g704-{kname:s}_{solvent[HEAVYWATER]}.png", dpi=100



print("############################ IAPWS R7-97 ############################")
Ts = np.asarray([-1.0, 25.0, 100.0, 200.0, 300.0, 360.0, 374.0])
Ts = Ts + 273.15
```

```
ps = pyiapws.psat(Ts)
for i in range(Ts.size):
    print(f"{Ts[i]:23.3f} K {ps[i]:23.3f} MPa.")

Ts = pyiapws.Tsat(ps)
for i in range(Ts.size):
    print(f"{Ts[i]:23.3f} K {ps[i]:23.3f} MPa.")

fig = plt.figure()
ax = fig.add_subplot()
ax.grid(visible=True, ls=':')
ax.set_xlabel("Ts /K")
ax.set_ylabel("ps /MPa")
Ts = np.linspace(0.0, 370.0, 500)
Ts = Ts + 273.15

ps = pyiapws.psat(Ts)
ax.plot(Ts, ps, "r-", label="ps(Ts)")

Ts = pyiapws.Tsat(ps)
ax.plot(Ts, ps, "b--", label="Ts(ps)")

ax.legend()
fig.savefig(f"../media/r797-r4.png", dpi=100, format="png")


T = 280.0 + 273.15
p = 8.0
res = pyiapws.wp(p, T, "v")*1000.0
print(f"v(8MPa,280°C) = {res:+23.16f} L/kg)")

x = np.asarray([8.0, 4.0, 6.0 ])
y = np.asarray([553.15, 1200.0, 2000.0])
r=pyiapws.wr(x, y)
s=pyiapws.wph(x, y)
print(r)
print(s)


plt.show()
```

SEE ALSO

iapws(1) **iapws_lib**(3) **iapws_psat**(3) **iapws_Tsat**(3) **iapws_kh**(3) **iapws_kd**(3) **iapws_kw**(3) **iapws_Tc**(3) **iapws_pc**(3) **ciaaw**(3), **codata**(3),