Introduction

============

Thermodynamic, hydrodynamic and non-linear modeling often requires thermodynamic derivatives, therefore IAPWS\_IF97 can calculate most property derivatives as functions of pressure and enthalpy, e.g.: dT/dp\_ph, cp\_ph, dv/dp\_ph and dv/dh\_ph. Since modeling often involves multiple dimensions that are discretized or meshed to form a set of either finite-difference or finite-element equations, IAPWS\_IF97 is vectorized even across regions (subcooled/compressed-liquid, saturated, superheated and supercritical). It is fully validated against the IAPWS\_IF97 standard test and is over 10X faster than XSteam-2.6.

Installation

============

[Download IAPWS\_IF97 from the MATLAB Central File Exchange](http://www.mathworks.com/matlabcentral/fileexchange/35710-iapwsif97-functional-form-with-no-slip), place on your path and use!

[![View IAPWS\_IF97 on File Exchange](https://www.mathworks.com/matlabcentral/images/matlab-file-exchange.svg)](https://www.mathworks.com/matlabcentral/fileexchange/35710-iapws\_if97)

IAPWS\_IF97 Standalone Function with No Slip

===========================================

IAPWS\_IF97 is 27 basic water functions of water properties, based on the International Association on Properties of Water and Steam Industrial Formulation 1997 (IAPWS-IF97), IAPWS-IF97-S01, IAPWS-IF97-S03rev, IAPWS-IF97-S04, IAPWS-IF97-S05, Revised Advisory Note No. 3 Thermodynamic Derivatives from IAPWS formulations 2008, Release on the IAPWS Formulation 2008 for the Viscosity of Ordinary Water Substance, 2008 Revised Release on the IAPWS Formulation 1985 for the Thermal Conductivity of Ordinary Water Substance.

Usage

-----

IAPWS\_IF97(FUN,IN1,IN2)

FUN is the desired function that may take 1 input, IN1, or 2 inputs, IN1 and IN2. IN1 and IN2 can be scalar, column vector or matrix, and IN1 and IN2 should be the same size. If a row vector is entered, it is transposed. If a scalar is entered for one input and the other input is a vector or matrix, then the scalar is repeated to form a vector or matrix of the same size as the other input.

FUN is a string that is formed by the property symbol, an underscore and the property symbols the function depends on. EG: 'k\_pT' is thermal conductivity, 'k', as a function of pressure, 'p', and temperature, 'T'. Derivatives are formed by prefixing 'd' to the property symbol and suffixing 'd' + the property symbol to which the derivative is with respect. EG: 'dTdp\_ph' is the derivative of temperature with respect to

pressure as a function of pressure and enthalpy, 'h', at constant enthalpy. The exception to this rule is 'cp\_ph' which is equivalent to '1/dTdh\_ph' or the reciprocal of the derivative of temperature with respect to enthalpy as a

function of pressure and enthalpy at constant pressure. All derivatives are with respect to pressure at constant enthalpy or v.v.

Saturation is indicated by suffixing 'sat', saturated liquid 'L' and saturated vapor 'V'.

`FUN = [d]<property-symbol>[sat|L|V][d<property-symbol>]\_<property-symbol>...`

Property Symbols:

\* `p [MPa]` pressure

\* `T [K]` temperature

\* `h [kJ/kg]` enthalpy

\* `v [m^3/kg]` specific volume the reciprocal of density, IE: v = 1/rho

\* `x` quality, mass fraction of liquid water in mixture

\* `k [W/m/K]` thermal conductivity

\* `mu [Pa\*s]` viscosity

\* `cp [kJ/kg/K]` specific heat at constant pressure

Basic functions:

`h\_pT`, `v\_pT`, `vL\_p`, `vV\_p`, `hL\_p`, `hV\_p`, `T\_ph`, `v\_ph`, `k\_pT`, `k\_ph`, `mu\_pT`, `mu\_ph`,

`dhLdp\_p`, `dhVdp\_p`, `dvdp\_ph`, `dvdh\_ph`, `dTdp\_ph`, `cp\_ph`, `dmudh\_ph`, `dmudp\_ph`,

`psat\_T`, `Tsat\_p`, `dTsatdpsat\_p`, `x\_ph`, `x\_hT`, `x\_pv`, `x\_vT`

Example:

>> press\_rng = logspace(-2,2,300); [MPa] pressure (p) range

>> temp\_rng = 273.15+linspace(1,800,300); [K] temperature (T) range

>> [p,T] = meshgrid(press\_rng,temp\_rng); [MPa,K] mesh p & T

>> h = IAPWS\_IF97('h\_pT',p,T); [kJ/kg] enthalpy = f(p,T)

>> psat = IAPWS\_IF97('psat\_T',temp\_rng); [MPa] saturation pressure

>> psat = psat(~isnan(psat)); trim out of range temperatures

>> hLsat = IAPWS\_IF97('hL\_p',psat); [kJ/kg] saturated liquid enthalpy

>> hVsat = IAPWS\_IF97('hV\_p',psat); [kJ/kg] saturated vapor enthalpy

>> pcrit = 22.064; [MPa] critical pressure

>> hLcrit = IAPWS\_IF97('hL\_p',pcrit);hVcrit = IAPWS\_IF97('hV\_p',pcrit);

>> Tcrit = IAPWS\_IF97('Tsat\_p',pcrit); hcrit = IAPWS\_IF97('h\_pT',pcrit,Tcrit);

>> hVL = hVsat - hLsat; [kJ/kg] heat of vaporization

>> hX = hLsat\*ones(1,9) + hVL\*(0.1:0.1:0.9); [kJ/kg] mixture enthalpy

Reference: [Revised IAPWS Industrial Formulation 1997](http://www.iapws.org/relguide/IF97-Rev.pdf)

Copyright (c) 2013 Mark Mifofski