
IAPWS-IF97: Auxiliary Functions for Region 3

A notebook for fitting functions to the phase boundary of IAPWS-IF97, region 3, in order to avoid iterative calculation of these phase boundaries. The data is normalized wrt the critical parameters and then transformed with the acos-function so that the infinite derivatives at the critical point are handled gracefully. apart from the usual approximation, several tricks are applied to make the approximation functions go exactly through the points where they border a neighbouring region, i.e. the critical point and the point $p(T=623.15)$, where region 3 meets region 1 and 2. Plots of the relative error of the fitted functions in per cent show their high accuracy.

pcrit,dcrit,Tcrit: critical data

plower: $p(T = 623.15)$ from the known $p(T)$ of IF97

dlmodel: model of the density on the boiling curve.

dvmodel: model of the density on the dew curve.

hlmodel: model of the enthalpy on the boiling curve.

hvmodel: model of the enthalpy on the dew curve.

Properties of water and steam in SI-units
in the Region 3 of the
IAPWS Industrial Formulation 1997
for the Thermodynamic Properties of Water and Steam

Copyright (C) 1999 Hubertus Tummescheit<hubertus@control.lth.se>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Reference:

Wagner, Wolfgang; Kruse, Alfred

Properties of water and steam;
the industrial standard IAPWS-IF97 for the thermodynamic
properties and supplementary equations for other
properties ; tables based on these equations

Berlin, Springer, 1998

Range of validity:

$T = 623.15 \dots 863.15 \text{ K}$
 $p = 1.0e6 \cdot (n1 + T \cdot (n2 + T \cdot n3)) \dots 100 \text{ MPa}$

$n1 = 0.34805185628969 \cdot 1e+3;$
 $n2 = -0.11671859879975 \cdot 1e+1;$
 $n3 = 0.10192970039326 \cdot 1e-2;$

The functions contain no input control, so make sure you are in region 3
(or close to it) before calling any of them.

The functions are tweaked so that the error at the borders to other areas of the IF97 is 0: they return exactly $dcrit$ if $p = p_{crit}$ and $hcrit$ if $p = p_{crit}$.

```
SetDirectory["/home2/hubertus/props/IAPWS/if97r3"]
Off[General::spell, General::spell1];
<< Formatting`Format`;
<< Formatting`Optimize`;

/home2/hubertus/props/IAPWS/if97r3
```

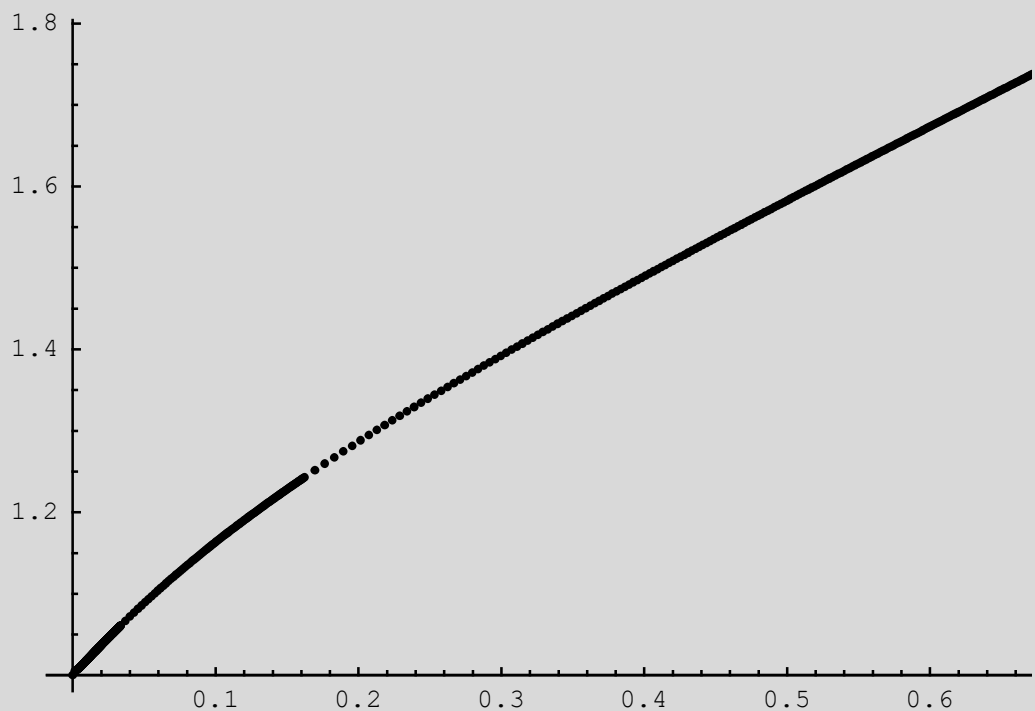
```
pcrit = 22064000;  
dcrit = 322;  
Tcrit = 647.096;  
hcrit = 2087546.84511715;  
scrit = 4412.021482236345;  
plower = 16529164.25260457;
```

Setting up the data:

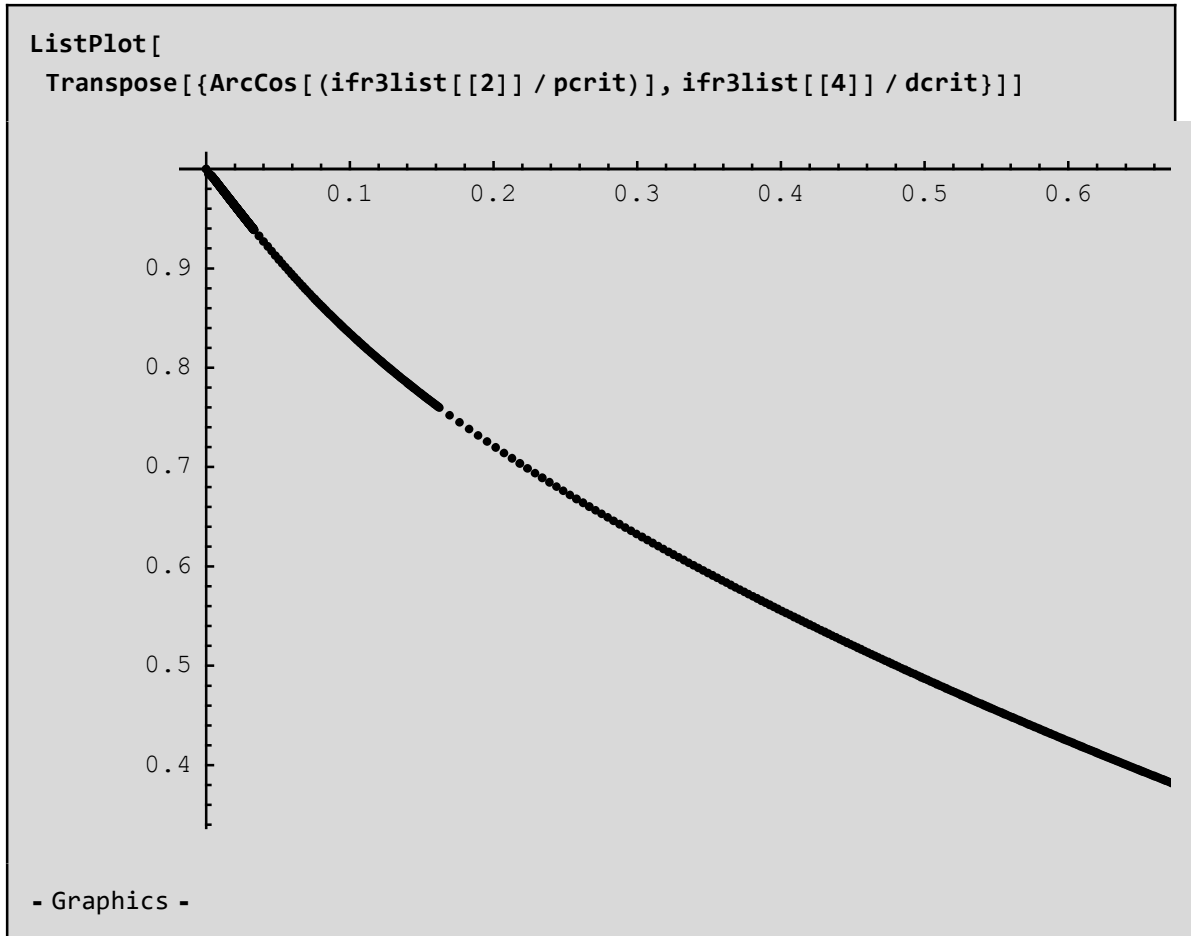
```
ifr3list = Transpose[ReadList["satp.dat", {Number, Number, Number,  
Number, Number, Number, Number, Number, Number}]]];
```

The plots show, how nicely ArcCos works to transform the data.

```
ListPlot[  
Transpose[{ArcCos[(ifr3list[[2]] / pcrit)], ifr3list[[3]] / dcrit}]]
```



- Graphics -



Approximation model for the liquid density

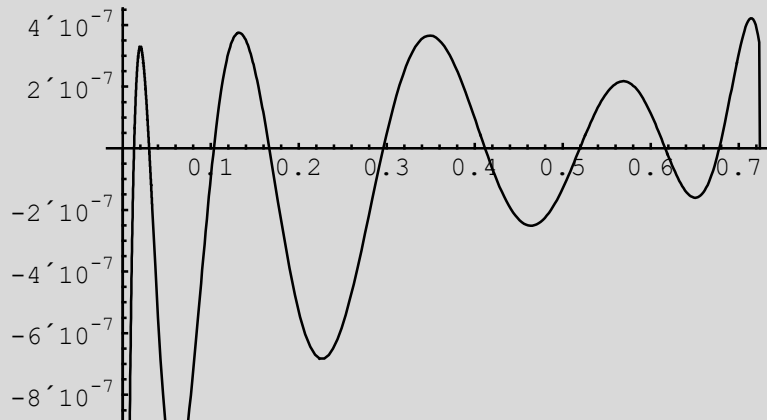
```
dllist =
  Transpose[{ArcCos[(ifr3list[[2]] / pcrit)], ifr3list[[3]] / dcrit - 1}];
dlmodel = Fit[dllist, {x, x^2, x^3, x^4, x^5, x^6, x^7, x^8, x^9}, x];
dlmodel = 1 + dlmodel;
dlerrlast = (dlmodel /. x -> dllist[[-1]][[1]]) - (dllist[[-1]][[2]] + 1);
dlxcoeff = -dlerrlast / dllist[[-1]][[1]]^10;
dlmodel = dlmodel + dlxcoeff x^10
dlerrlast = (dlmodel /. x -> dllist[[-1]][[1]]) - (dllist[[-1]][[2]] + 1);
```

$$1 + 1.90322 x - 2.53149 x^2 - 8.19145 x^3 + 94.342 x^4 - 369.368 x^5 + 796.663 x^6 - 994.539 x^7 + 673.258 x^8 - 191.431 x^9 + 0.000525365 x^{10}$$

```

compdmodel = Transpose[{Transpose[dllist][[1]],
  ((dlmodel /. x -> Transpose[dllist][[1]]) dcrit -
    (Transpose[dllist][[2]] + 1) dcrit) /
  ((Transpose[dllist][[2]] + 1) dcrit * 100)}];
ListPlot[compdmodel, PlotJoined -> True]

```



- Graphics -

Approximation model for the vapor density

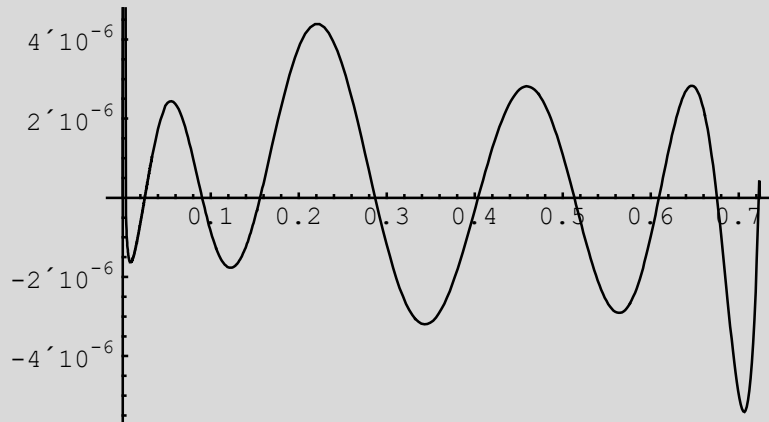
```

dvlist =
  Transpose[{ArcCos[ifr3list[[2]] / pcrit], ifr3list[[4]] / dcrit - 1}];
dvmodel = Fit[dvlist, {x, x^2, x^3, x^4, x^5, x^6, x^7, x^8, x^9}, x];
dvmodel = 1 + dvmodel;
dverrlast = (dvmodel /. x -> dvlist[[-1]][[1]]) - (dvlist[[-1]][[2]] + 1);
dvxcoeff = -dverrlast / dvlist[[-1]][[1]]^10;
dvmodel = dvmodel + dvxcoeff x^10

```

$$1 - 1.84639 x - 1.14479 x^2 + 59.187 x^3 - 403.539 x^4 + 1437.2 x^5 - 3015.85 x^6 + 3740.58 x^7 - 2537.38 x^8 + 725.876 x^9 - 0.00111511 x^{10}$$

```
compdmodel = Transpose[{Transpose[dvlist][[1]],  
  ((dmodel /. x -> Transpose[dvlist][[1]]) dcrit) -  
  (Transpose[dvlist][[2]] + 1) dcrit) /  
  ((Transpose[dvlist][[2]] + 1) dcrit * 100)}];  
ListPlot[compdmodel, PlotJoined -> True]
```



- Graphics -

```
(dlmodel /. x -> ArcCos[1.0]) dcrit
```

322.

Approximation model for the liquid enthalpy

```

hllist =
  Transpose[{ArcCos[ifr3list[[2]] / pcrit], ifr3list[[5]] / hcrit - 1}];
hlmodel = Fit[hllist, {x, x^2, x^3, x^4, x^5, x^6, x^7}, x];
hlmodel = 1 + hlmodel;
hlerrlast = (hlmodel /. x -> hllist[[-1]][[1]]) - (hllist[[-1]][[2]] + 1);
hlxcoeff = -hlerrlast / hllist[[-1]][[1]]^8;
hlmodel = hlmodel + hlcoeff x^8
hlerrlast = (hlmodel /. x -> hllist[[-1]][[1]]) - (hllist[[-1]][[2]] + 1);

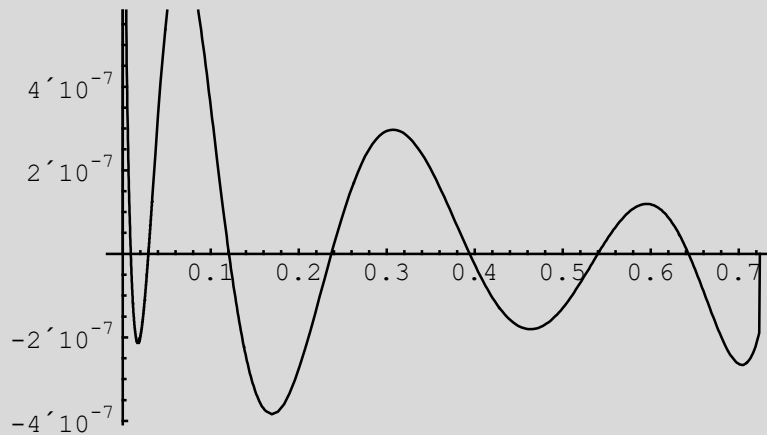
```

$$1 - 0.494559 x + 1.3468 x^2 - 3.88939 x^3 + 6.67939 x^4 - 6.7582 x^5 + 3.55892 x^6 - 0.717982 x^7 - 0.000115203 x^8$$

```

comphlmodel = Transpose[{Transpose[hllist][[1]],
  ((hlmodel /. x -> Transpose[hllist][[1]]) hcrit -
  (Transpose[hllist][[2]] + 1) hcrit) /
  ((Transpose[hllist][[2]] + 1) hcrit * 100)}];
ListPlot[comphlmodel, PlotJoined -> True]

```



- Graphics -

Approximation model for the vapor enthalpy

```

hvlist =
  Transpose[{ArcCos[(ifr3list[[2]] / pcrit)], ifr3list[[6]] / hcrit - 1}];
hvmodel = Fit[hvlist, {x, x^2, x^3, x^4, x^5, x^6, x^7}, x];
hvmodel = 1 + hvmodel;
hverrlist =
  (hvmodel /. x -> hvlist[[-1]][[1]]) - (hvlist[[-1]][[2]] + 1);
hvxcoeff = -hverrlist / hvlist[[-1]][[1]]^8;
hvmodel = hvmodel + hvxcoeff x^8
hverrlist = (hvmodel /. x -> hvlist[[-1]][[1]]) - (hvlist[[-1]][[2]] + 1);

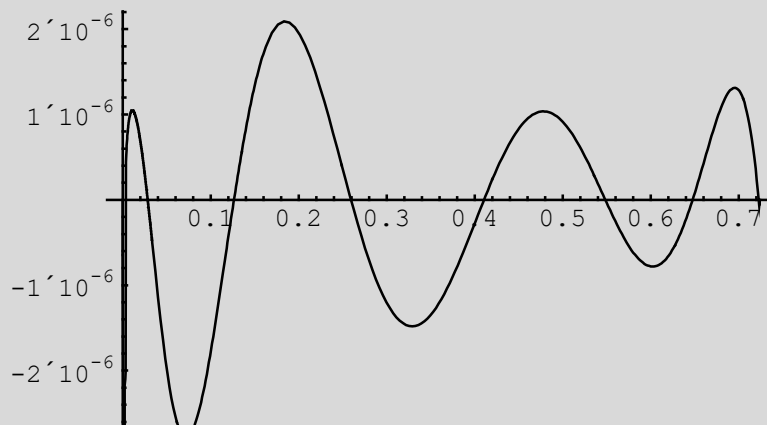
1 + 0.488015 x + 0.207967 x^2 - 6.08412 x^3 + 25.0889 x^4 -
48.3822 x^5 + 45.6649 x^6 - 16.9856 x^7 + 0.000661694 x^8

```

```

comphvmodel = Transpose[{Transpose[hvlist][[1]],
  ((hvmodel /. x -> Transpose[hvlist][[1]]) hcrit -
  (Transpose[hvlist][[2]] + 1) hcrit) /
  ((Transpose[hvlist][[2]] + 1) hcrit * 100)}];
ListPlot[comphvmodel, PlotJoined -> True]

```



- Graphics -

This can be used for implementation, if desired


```
CForm[Coefficient[dlmodel, x, {0, 1, 2, 3, 4, 5, 6, 7, 8}]]
```

```
List(1,1.903224078789084,-2.5314861672948212,-8.191449508347372,
      94.34196243818363,-369.367688229012,796.6628020044742,-994.5385527404217,
      673.2581279257353)
```

```
CForm[Horner[dlmodel]]
```

```
1 + x*(1.903224078789084 + x*(-2.5314861672948212 +
      x*(-8.191449508347372 +
      x*(94.34196243818363 +
      x*(-369.367688229012 +
      x*(796.6628020044742 +
      x*(-994.5385527404217 +
      x*(673.2581279257353 +
      (-191.43077640286174 + 0.0005253653213400976*x)*x)
      )))
```

```
CForm[Horner[dvmodel]]
```

```
1 + x*(-1.8463850806182265 + x*(-1.1447872580856484 +
      x*(59.18702182102936 +
      x*(-403.5391416771118 +
      x*(1437.2007185993543 +
      x*(-3015.853526658207 +
      x*(3740.5790166102206 +
      x*(-2537.375804085372 +
      (725.876193619163 - 0.0011151121505620263*x)*x)
      )))
```

```
CForm[Horner[h1model]]
```

```
1 + x*(-0.4945586958015183 + x*(1.3468000161552502 +
      x*(-3.889388149672598 +
      x*(6.679385458806799 +
      x*(-6.758202382250221 +
      x*(3.5589197163769146 +
      (-0.7179818444620986 - 0.00011520330341994825*x)*x)
      )))
```

```
CForm[Horner[hvmodel]]
```

```
1 + x*(0.4880153718887746 + x*(0.20796707404989687 +
      x*(-6.084122693550026 +
        x*(25.088876003770633 +
          x*(-48.38215176432967 +
            x*(45.664891610344206 +
              (-16.985554414847027 + 0.0006616936574471414*x)*x))))))
```

Approximation model for the liquid entropy

```
sllist =
  Transpose[{ArcCos[(ifr3list[[2]] / pcrit)], ifr3list[[7]] / scrit - 1}];
slmodel = Fit[sllist, {x, x^2, x^3, x^4, x^5, x^6, x^7}, x];
slmodel = 1 + slmodel;
slerrlast = (slmodel /. x -> sllist[[-1]][[1]]) - (sllist[[-1]][[2]] + 1);
slxcoeff = -slerrlast / sllist[[-1]][[1]]^8;
slmodel = slmodel + slxcoeff x^8
slerrlast = (slmodel /. x -> sllist[[-1]][[1]]) - (sllist[[-1]][[2]] + 1)
```

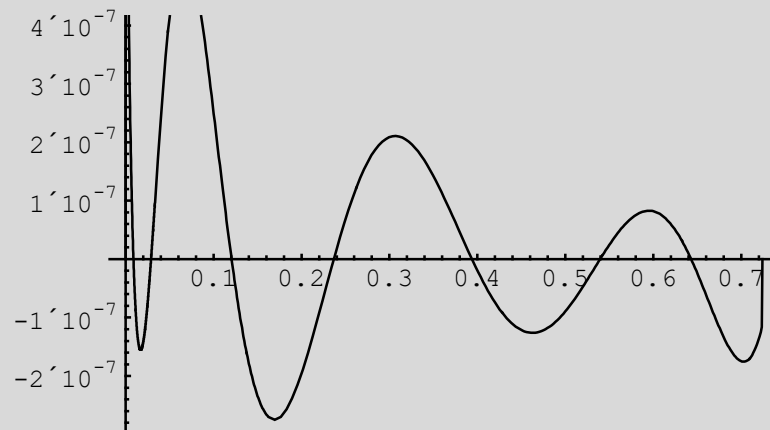
```
1 - 0.361607 x + 0.996278 x^2 - 2.85955 x^3 + 4.9063 x^4 -
  4.97409 x^5 + 2.62497 x^6 - 0.531995 x^7 - 0.00080645 x^8
```

```
0.
```

```

compmodel = Transpose[{Transpose[sllist][[1]],
  ((smodel /. x -> Transpose[sllist][[1]]) scrit -
    (Transpose[sllist][[2]] + 1) scrit) /
    ((Transpose[sllist][[2]] + 1) scrit * 100)}];
ListPlot[compmodel, PlotJoined -> True]

```



- Graphics -

Approximation model for the vapor entropy

```

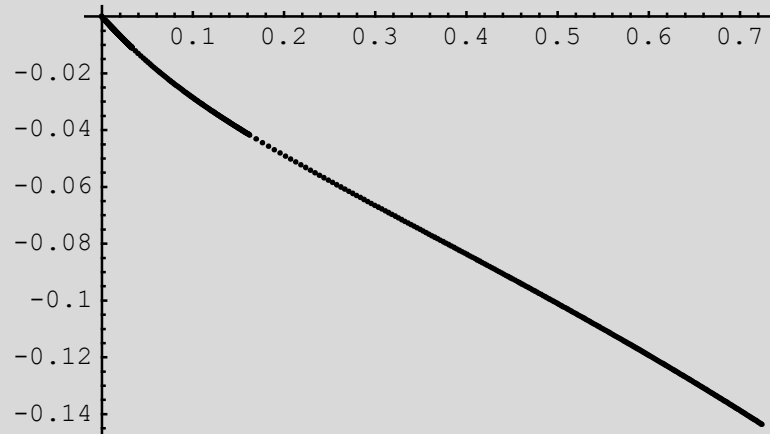
svlist =
  Transpose[{ArcCos[ (ifr3list[[2]] / pcrit)], ifr3list[[8]] / scrit - 1}];
svmodel = Fit[svlist, {x, x^2, x^3, x^4, x^5, x^6, x^7}, x];
svmodel = 1 + svmodel;
sverrlast =
  (svmodel /. x -> svlist[[-1]][[1]]) - (svlist[[-1]][[2]] + 1);
svxcoeff = -sverrlast / svlist[[-1]][[1]]^8;
svmodel = svmodel + svxcoeff x^8
sverrlast = (svmodel /. x -> svlist[[-1]][[1]]) - (svlist[[-1]][[2]] + 1)

```

$$1 + 0.356826 x + 0.164246 x^2 - 4.42535 x^3 + 18.3245 x^4 - 35.3386 x^5 + 33.3618 x^6 - 12.4087 x^7 + 0.000481005 x^8$$

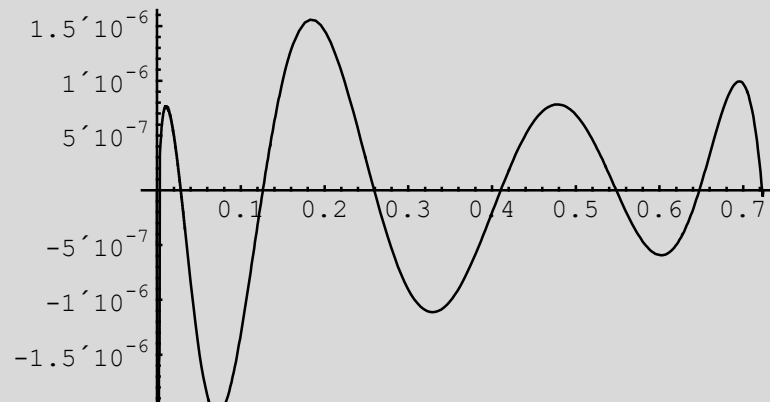
0.

```
ListPlot[
  Transpose[{ArcCos[(ifr3list[[2]]/pcrit)], ifr3list[[7]]/scrit - 1}]]
```



- Graphics -

```
compsvmodel = Transpose[{Transpose[svlist][[1]],
  ((svmodel /. x -> Transpose[svlist][[1]]) scrit -
  (Transpose[svlist][[2]] + 1) scrit) /
  ((Transpose[svlist][[2]] + 1) scrit * 100)}];
ListPlot[compsvmodel, PlotJoined -> True]
```



- Graphics -

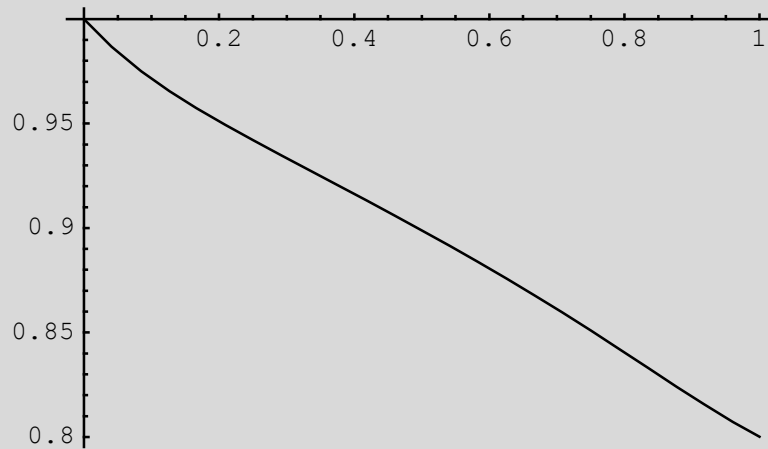
CForm[Horner[s1model]]

```
1 + x*(-0.36160692245648063 + x*
  (0.9962778630486647 + x*(-2.8595548144171103 +
    x*(4.906301159555333 +
      x*(-4.974092309614206 +
        x*(2.6249651699204457 +
          (-0.5319954375299023 - 0.00008064497431880644*x)*x))))))
```

CForm[Horner[svmodel]]

```
1 + x*(0.35682641826674344 + x*(0.1642457027815487 +
  x*(-4.425350377422446 +
    x*(18.324477859983133 +
      x*(-35.338631625948665 +
        x*(33.36181025816282 +
          (-12.408711490585757 + 0.0004810049834109226*x)*x))))))
```

Plot[s1model, {x, 0, 1}]



- Graphics -