

# Files of module SAHA-S3 for CESAM2k

## 1. Equation of state SAHA-S3

V.K.Gryaznov, I.L.Iosilevsky, A.N.Starostin  
Version 3 of 9 Mar 2011.

Original tables and their detailed description can be downloaded by the following link:  
[Download EOS tables \(38.2 MB\)](#).

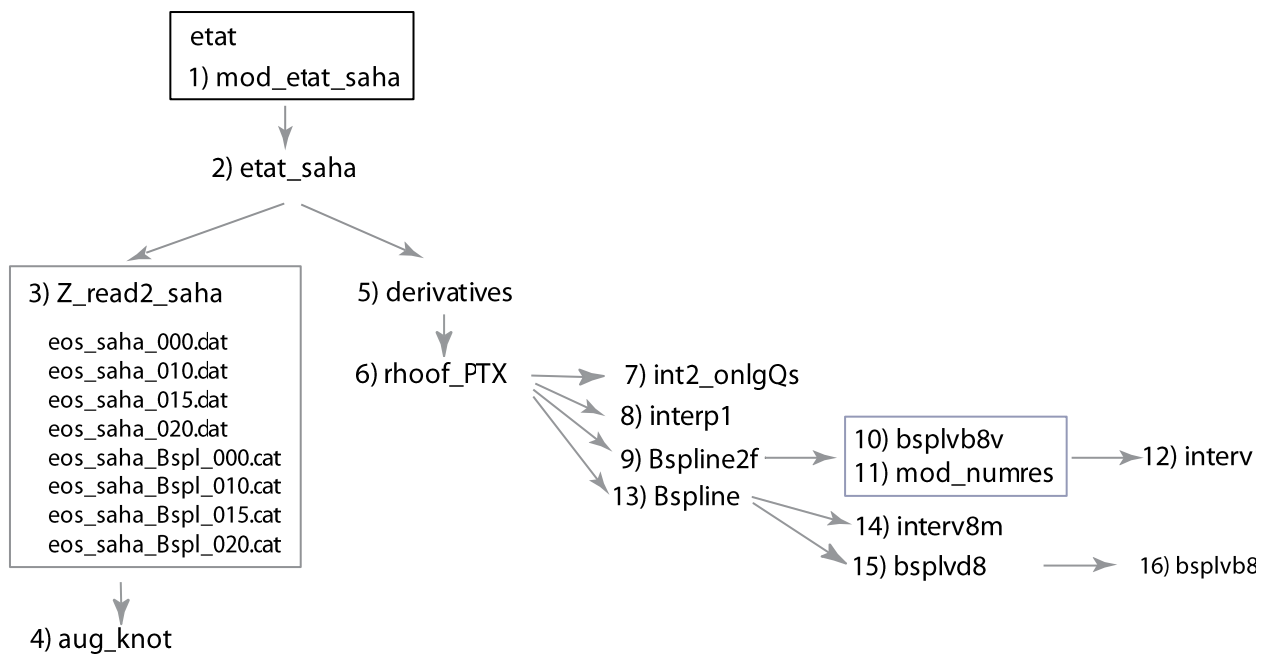
The tables are rectangular and have mesh evenly spaced in  $\log_{10}(T)$  and  $\log_{10}(Q)$ , here  $Q_s = \rho/T6^{2.25}$ ,  $T6=T/10^6$ . There are no gaps in the tables. Dimensions are:

$\log_{10}(T)$ : 3.20 ... 7.55, step 0.025  
 $\log_{10}(Q_s)$ : -4.50 ... 0.45, step 0.05  
X: 0.1 ... 0.9, step 0.1  
Z: 0.010, 0.015, 0.020

Please cite the following paper when using SAHA-S3:

V.K.Gryaznov, S.V.Ayukov, V.A.Baturin, I.L.Iosilevskiy, A.N.Starostin, V.E.Fortov. Solar plasma: calculation of thermodynamic functions and equation of state. Journal of Physics A: Mathematical and General, v.39, 4459-4464, 2006.  
DOI: [10.1088/0305-4470/39/17/S22](https://doi.org/10.1088/0305-4470/39/17/S22)

## 2. Scheme of SAHA-S3 module for CESAM2k



### 3. Brief description of the files

1) **mod\_etat\_saha** Purpose of the module is working with SAHA-S3 EOS tables in sense of CESAM2k architecture.

2) **etat\_saha**(pp,tt,xchim,deriv,  
ro,drop,drot,drox,  
u,dup,dut,dux,  
delta,deltap,deltat,deltax,  
cp,dcpp,dcpt,dcpx,  
gradad,dgradadp,dgradadt,dgradadx,  
alfa,beta,gamma1)

Interface of the equation of state SAHA-S3 with CESAM2k.

Computations are performed for current metallicity Z which depends on space coordinate and time.

Input:

p : pressure  
t : temperature  
xchim : chemical composition  
deriv=.FALSE. : avoide computations of some derivatives

Output:

ro : density and derivatives  
u : internal energy and derivatives  
delta :  
cp : specific heat and derivatives  
gradad : adiabatic gradient and derivatives  
alpha  
beta  
gamma1

### 3) Z\_read2\_saha

The subroutine sets the two values of metallicity Ztab1 and Ztab2 which are used for interpolation. By default, Ztab1=0.010 and Ztab2=0.015 which is suitable for the solar model. They can be changed in the text of the code. Possible values are 0.000, 0.010, 0.015 and 0.020.

Then it reads SAHA-S3 data tables eos\_saha\_\*\*\*.dat, prepared for CESAM; the result is written in xz1(mx,nt,nqs,nclm) and xz2(mx,nt,nqs,nclm).

After that, it reads SAHA\_S3 tables of B-spline coefficients eos\_saha\_Bspl\_\*\*\*.dat; the result is written in Beos1(iqs,it,imv,ix) and Beos2(iqs,it,imv,ix).

An initialization of knots for X, lgT and lgQs is also performed.

Files eos\_saha\_\*\*\*.dat and eos\_saha\_Bspl\_\*\*\*.dat are saved in the directory CESAM-DATA.

Asterisks \*\*\* in the names of files denote metallicity  $Z$  for the tables: 000  $Z=0$ ; 010  $Z=0.010$ ; 015  $Z=0.015$ ; 020  $Z=0.020$ .

It is important to change the names of files for reading if Ztab1 and Ztab2 have been changed!

Sense of variables in a line of table eos\_saha\_\*\*\*.dat:

- (1) P, dyne/cm<sup>2</sup>
- (2) u, egr/g
- (3) du/drho, CGS
- (4) {du/dT}\_P=cp, CGS
- (5) {dlgP/dlgrho}\_T=chi\_rho
- (6) delta
- (7) Gamma1
- (8) grad\_ad
- (9) Gamma3-1
- (10) rho, g/cm<sup>3</sup>
- (11) {dlgP/dlgX}\_T,rho
- (12) {du/dx}\_T,rho, CGS

#### 4) aug\_knot(X,knots)

Create knots for Bspline 1D interpolation.

#### 5) derivatives(pp,tt,xchim,deriv, ro,drop,drot,drox, u,dup,dut,dux, delta,deltap,deltat,deltax, cp,dcpp,dcpt,dcpx, gradad,dgradadp,dgradadt,dgradadx, alfa,beta,gamma1)

Compute density and some other functions and their derivatives for metallicity  $Z$  from the SAHA-S3 tables ( $Z=0.000, 0.010, 0.015, 0.020$ ).

Input:

p : pressure  
t : temperature  
xchim : chemical composition  
deriv=.FALSE. : avoide computations of some derivatives

Output:

ro : density and derivatives  
u : internal energy and derivatives  
delta :  
cp : specific heat and derivatives  
gradad : adiabatic gradient and derivatives  
alpha  
beta  
gamma1

## 6) rhoof\_PTX(lgT,lgP,X, eos, deos)

Give density rho as function of pressure P, temperature T, and hydrogen abundance X according to SAHA-S3 equation of state. It is equivalent to rhoofp from CESAM2k etat\_opalZ code.

Input: lgT,lgP,X

Output: eos[1:12] – massive of 12 interpolated values for given lgT,lgP,X:

```
eos(1) = P, dyne/cm^2
eos(2) = u, egr/g
eos(3) = du/drho, CGS
eos(4) = {du/dT}_P=cp, CGS
eos(5) = {dlgP/dlgrho}_T=chi_rho
eos(6) = delta
eos(7) = Gamma1
eos(8) = grad_ad
eos(9) = Gamma3-1
eos(10) = rho, g/cm^3
eos(11) = {dlgP/dlgX}_T,rho
eos(12) = {du/dx}_T,rho, CGS
```

deos[1:12,1:3] – derivatives of these functions:

```
deos[1:12,1] - d eos/dlgT ,
deos[1:12,2] - d eos/dlgQs ,
deos[1:12,3] - d eos/dX .
```

## 7) int2\_onlgQs(X,lgT,P\_line,rho\_line)

2D LINEAR INTERPOLATION of pressure P and density rho for given hydrogen abundance X, logarithm of temperature lgT and for all net-points lgQs from SAHA-S3 tables.

Input: X, lgT.

Output: P\_line[1:100] - array of lgP values,  
rho\_line[1:100] – array of lgrho values.

## 8) interp1(X\_line, X\_line,X,Y)

Just 1D linear interpolation.

Input: X\_line[:] – mesh points,  
Y\_line[:] – function values in mesh points,  
X – input point.

Output: Y – interpolated value of function in point X.

### 9) **Bspline2f**(X0, lgT0, lgQs0, Ys)

B-spline interpolation of SAHA-S3 tables for given X0, lgT0, lgQs0.

**Bspline2f** computes only two values Ys(1) and Ys(5) instead of all 12 (as **Bspline** does).

Input: X0, lgT0, lgQs0.

Output: Ys(1) = P,

Ys(5) = {d<sub>lgP/d<sub>lg</sub>rho</sub>}\_T=chi\_rho.

### 10) **bsplvb8v**(T, Jhigh, x, LC, BIATX)

Values of all (needed) B-splines, nonzero in point x, with order Jhigh simplified in respect of index knots of B-spline in array T.

Original from: C. de Boor. "A Practical Guide To Splines. Chapter X".

REWRITTEN by V.Baturin, 2015 for purpose of not-a-knot interpolation with MATLAB.

Input:

T[:] - knots, length = Left+Jhigh, non-decreasing order;

x - point where B-spline is computed;

Left - integer, T(Left) .LE. T(Left+1).

Output:

BIATX - array with Jout length, BIATX(i) contains value of polynom order Jout in point x (B-spline B(Left-Jout+i), Jout, T) on interval (T(Left), T(Left+1)).

### 11) module **mod\_numres**

contains

FUNCTION **locate**(xx,x)

Find left index of interval from array xx[:] which includes point x.

### 12) **interv**(xt, lxt, x, left, mflag)

From "A practical guide to splines" by C. de Boor.

Computes left = max( i : xt(i) .lt. xt(lxt) .and. xt(i) .le. x ).

### 13) **Bspline**(X0, lgT0, lgQs0, Ys, dYs)

B-spline interpolation of SAHA-S3 tables for given X0, lgT0, lgQs0.

**Bspline** computes all 12 values Ys instead of two (as **Bspline2f** does).

**Bspline** computes also derivatives dYs of some these values with respect to X, lgT, lgQs.

Input:  $X_0$ ,  $\lg T_0$ ,  $\lg Q_{s0}$ .

$Ys[1:12]$  – massive of 12 interpolated values for given  $\lg T, \lg P, X$ :

$Ys(1)$  =  $P$ , dyne/cm<sup>2</sup>  
 $Ys(2)$  =  $u$ , egr/g  
 $Ys(3)$  =  $du/drho$ , CGS  
 $Ys(4)$  =  $\{du/dT\}_{P=cp}$ , CGS  
 $Ys(5)$  =  $\{d\lg P/d\lg rho\}_{T=chi\_rho}$   
 $Ys(6)$  =  $\delta$   
 $Ys(7)$  =  $\Gamma_1$   
 $Ys(8)$  =  $\text{grad}_{ad}$   
 $Ys(9)$  =  $\Gamma_3-1$   
 $Ys(10)$  =  $\rho$ , g/cm<sup>3</sup>  
 $Ys(11)$  =  $\{d\lg P/d\lg X\}_{T,\rho}$   
 $Ys(12)$  =  $\{du/dx\}_{T,\rho}$ , CGS

$dYs[1:12,1:3]$  – derivatives of these functions:

$dYs[1:12,1]$  -  $dYs/d\lg T$  ,  
 $dYs[1:12,2]$  -  $dYs/d\lg Q_s$  ,  
 $dYs[1:12,3]$  -  $dYs/dX$  .

#### 14) `interv8m` ( $xt$ , $x$ , $left$ , $mflag$ )

From “A practical guide to splines” by C. de Boor.

Computes  $left = \max(i : xt(i) .lt. xt(1xt) .and. xt(i) .le. x )$ .

#### 15) `bsplvd8` ( $t$ , $k$ , $x$ , $left$ , $dbiatx$ , $nderiv$ )

From “A practical guide to splines” by c. de Boor (7 may 92).

REWRITTEN in double precision and fortran 90 by Baturin, 2015

Calculates value and derivatives of all b-splines which do not vanish at  $x$ .

! calls `bsplvb8`

Input:

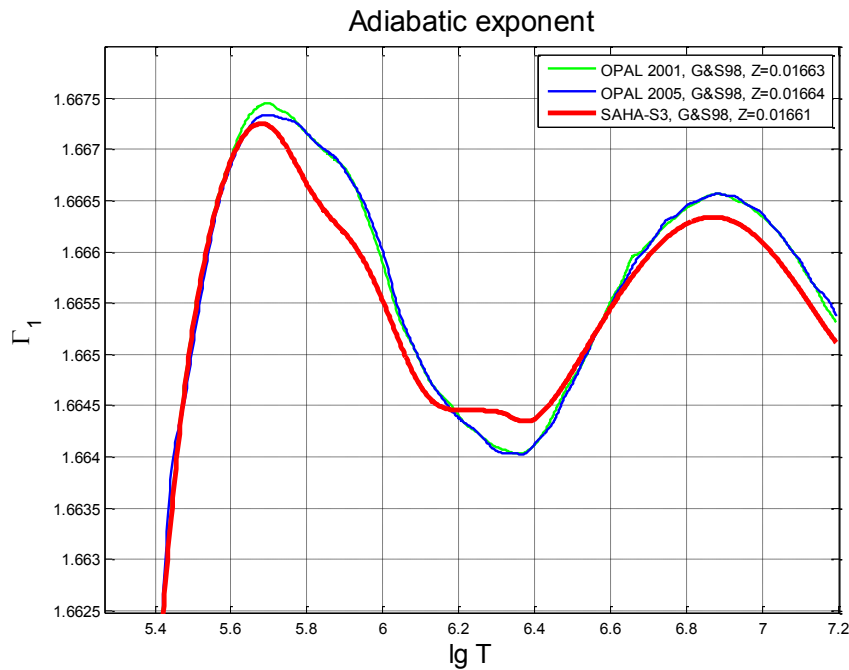
$t$  the knot array, of length  $left+k$  (at least)  
 $k$  the order of the b-splines to be evaluated (power + 1)  
 $x$  the point at which these values are sought  
 $left$  an integer indicating the left endpoint of the interval of interest. The  $k$  b-splines whose support contains the interval  $(t(left), t(left+1))$  are to be considered.  
 $nderiv$  an integer indicating that values of b-splines and their derivatives up to but not including the  $nderiv$ -th are asked for.

Output:  
dbiatx an array of order  $(k, \text{nderiv})$ . Its entry  $(i, m)$  contains value of  $(m-1)$ st derivative of  $(\text{left}-k+i)$ -th b-spline of order  $k$  for knot sequence  $\mathbf{t}$ ,  $i=1, \dots, k$ ,  $m=1, \dots, \text{nderiv}$ .

#### 4. Comparison of SAHA-S3 and other equations of state

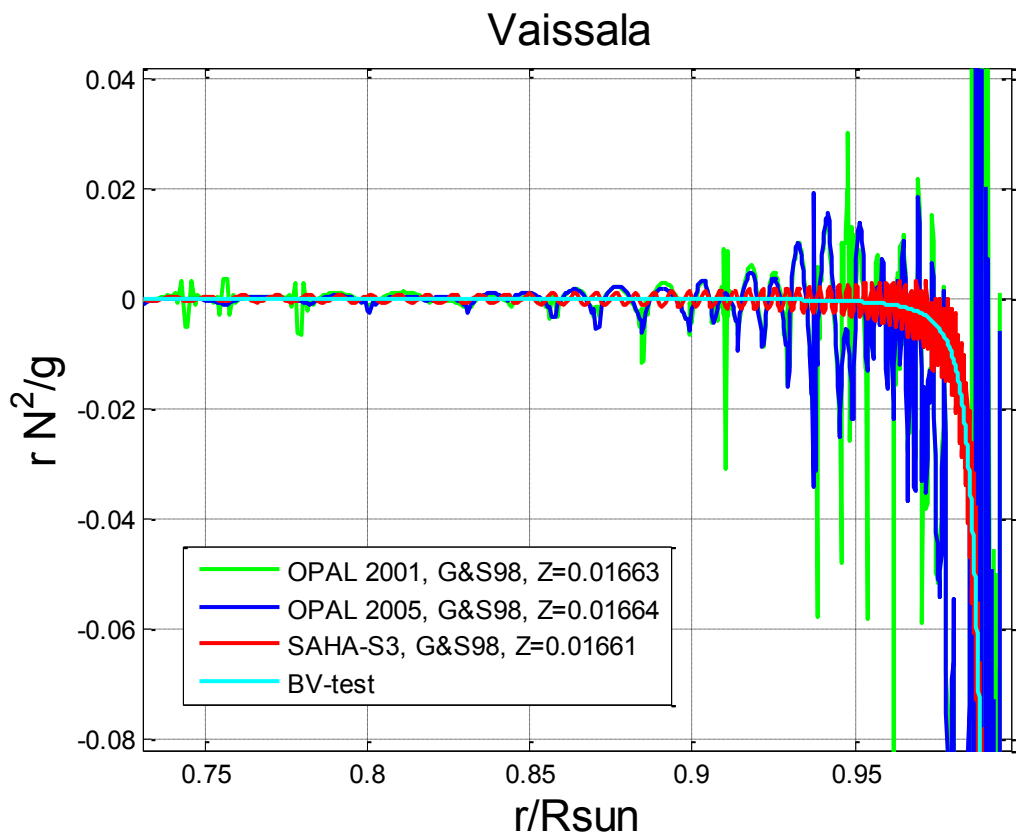
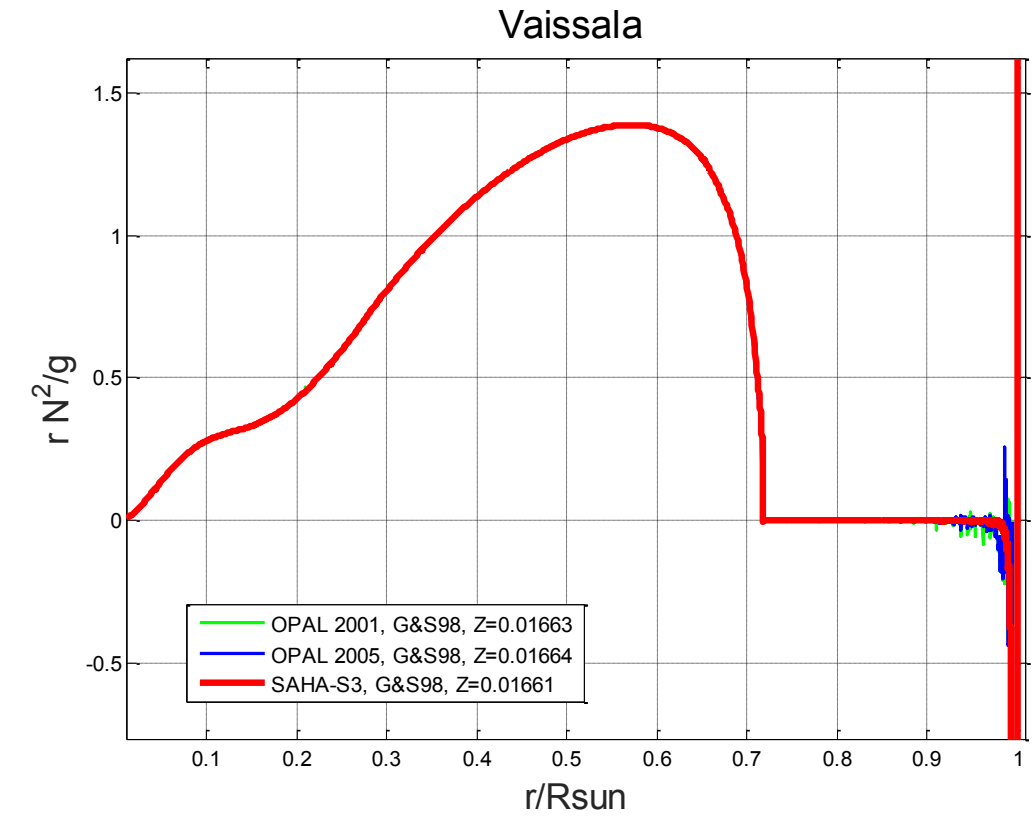
We present some results which are interesting for helioseismology. All the figures illustrate profiles of physical values in calibrated models of the Sun. These models differ by versions of equation of state. We have used OPAL 2001, which had been already included in CESAM2k. To incorporate OPAL 2005 we substituted tables in the CESAM2k in the interpolation code. SAHA-S3 equation of state has been incorporated in CESAM2k with the rewritten interpolation code. The new interpolation code is more effective and smooth. Additionally, SAHA-S3 EOS allows metallicity interpolation on every step of space and time.

**4.1. Adiabatic exponent  $\Gamma_1$**  in different models of the present Sun is plotted for the deep part of solar interiors. The surface part is not shown. In those levels, deviations in  $\Gamma_1$  can achieve up to  $1e-3$  and it can be resolved by helioseismology.

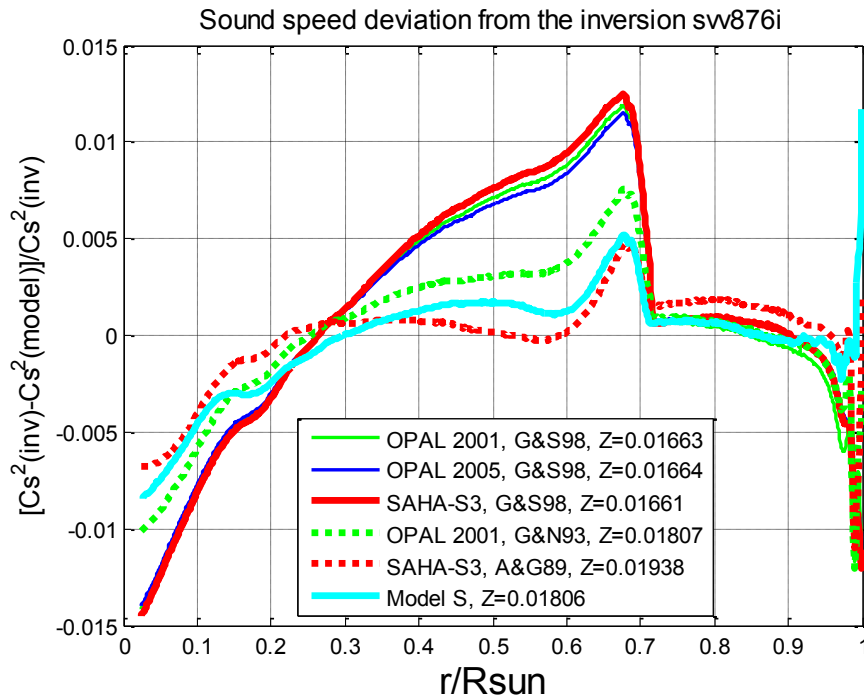




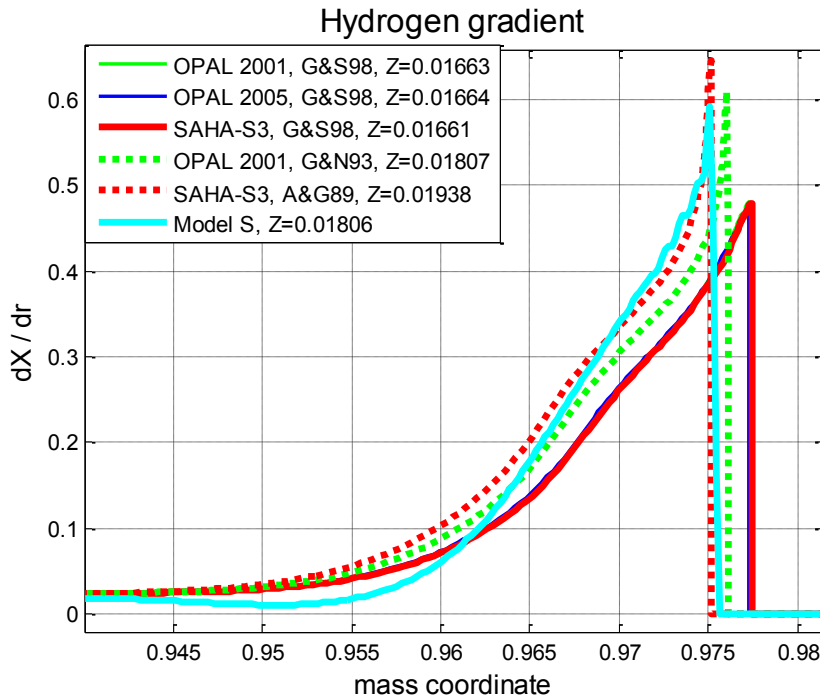
**4.2. Buoyancy frequency** in the SAHA-S3 equation of state has smaller noise than in OPALs. There are some ways to improve the situation even more (see cyan curve on the bottom panel).



**4.3. Sound speed** deviation from the inversion shows strong dependence on metallicity  $Z$ . There are also some other features which are interesting to study. Such curves are good tools for comparing theory and observations.



**4.4. Hydrogen gradient** below the bottom of the convection zone reveals the difference of diffusion computations in CESAM2k and Model S. Understanding the causes of these discrepancy seems very interesting and promising for further improving CESAM2k code . Three solid curves (for low  $Z$ ) coincide on the figure.



## 5. Conclusion

Equation of state SAHA-S3 is included in CESAM2k code. First results of solar modeling are obtained and compared with previous computations. There are interesting perspectives for further improving the code and for solving helio- and astro-seismic tasks.