Notes on Opacity-Interpolationroutine OPINT (v11, 05/03/2006)

1. Introduction

This package interpolates opacities in ρ,T,X and Z (ρ = density, T = temperature, X = hydrogen mass fraction, Z = heavy elements mass fraction). In this package the latest OPAL-tables [RI95] are used. One can choose between two interpolation algorithms. The first algorithm (minimum norm) uses a C^1 interpolant defined in piecewise fashion over triangles in the $\rho - -T$ plane [MC89] [Nie83]. The global interpolant is constucted by a nine paramter version of a finite interpolation scheme introduced by Nielson [Nie80]. The second algorithm uses birational splines and is implemented according to Helmuth Spath [Spa91]. Interpolation in X and Z is performed using Akima's univariate interpolation scheme [Aki70, Aki91]. For the low-temperature opacites, tables from Alexander & Ferguson (1994, ApJ 437, 879) [AF94] and from Alexander & Ferguson (2005, ApJ 623, 585) [AF05] are available.

Electron conduction (EC) according to Itoh et al. (1983, ApJ 273, 774) can be enabled for all tables. EC could be important in more massive stars. If EC is enabled, the EC becomes effective (i.e. it is 'switched' on) if the EC contribution modifies the resulting opcacity by $\geq 10^{-5}$. This treshold can be changed in the code (dopints.f, dopintc.f, dopintf.f) by changing the currently defined parameter **econ=5.0d0** to any other value.

2. Installation

2.1 Using the provided opacity tables

The following five tables are provided for 'little_endian' machines (such as machines with intel processors; to generate tables for 'big_endian' machines see Section 2.2; for the opacity table 1), given below, also 'big_endian' binary files are provided):

Opacity table	Abundance	Description
1) OPAL95+AF94	GN93	Opal (1995) tables plus Alexander-Furgenson (1994) tables using the Grevesse-Noel (1993) abundances.
2) OPAL95+AF05	GN93	Opal (1995) tables plus Alexander-Furgenson (2005) tables using the Grevesse-Noel (1993) abundances.
3) OPAL95+AF05	AGS05	Opal (1995) tables plus Alexander-Furgenson (1994) tables using the Asplund-Grevesse-Sauval (2005) abundances.
4) OPAL95+AF05	AGS05pNe45Ar40	As table 3), but with Neon enhanced by 0.45 dex and Argon by 0.40 dex.
5) OPAL95+AF05	AGS05pNe40	As table 3), but with Neon enhanced by 0.50 dex.

Table 1: Provided opacity tables

To use any of the tables listed above, copy in the main-directory the appropriate Makefile_*, listed in the table below, to the filename 'Makefile', e.g.: to use table 3) (OPAL95+AF05 and AGS05 abundances) say:

cp Makefile_AGS05 Makefile

Table (Abundance)	Makefile	OPINTPATH
OPAL95+AF94 (GN93)	Makefile_GN93_AF94	OPINTPATH_GN93_AF94
OPAL95+AF05 (GN93)	Makefile_GN93	OPINTPATH_GN93
OPAL95+AF05 (AGS05)	Makefile_AGS05	OPINTPATH_AGS05
OPAL95+AF05 (AGS05pNe45Ar40)	Makefile_AGS05pNe45Ar40	OPINTPATH_AGS05
OPAL95+AF05 (AGS05pNe40)	Makefile_AGS05pNe50	OPINTPATH_AGS05

Table 2: Provided Makefiles and OPINTPATHs

Next modify the **Makefile** with an editor (such as vi) according to your compiler needs, such as compiler-flages, ranlib, etc and execute the command

make lib

which builds the library: './lib/libopint.a'.
Next copy the appropriate OPINTPATH_* file to <appl_path>/OPINTPATH, where <appl_path> is the directory name of your application for which you wish to use this interpolation package, e.g.:

cp OPINTPATH_AGS05 <appl_path>/OPINTPATH

With an editor (such as vi) modify the pathes of the opacity tables in the file **OPINTPATH**, i.e. change the pathes to the (e.g., absolute) installation pathes <inst_path> where you installed the opacity package, as demonstrated below in Table 3 for the OPAL95+AF05 (AGS05) opacity tables.

Entry #	pathes of opacity tables	description
1	<inst_path>/v11/ol95_AGS05/little_endian/opalxe.bin	Opacity values of OPAL tables
2	<inst_path>/v11/af05_AGS04/little_endian/af05xe.bin	Opacity values of Alexander-Furgenson tables
3	<inst_path>/v11/ol95_AGS05/little_endian/opalxeAF05-pd.bin	Partial derivatives of OPAL and Alexander-Furgenson tables
4	<inst_path>/v11/ol95_AGS05/little_endian/ival95.dat	used for determinig extrapolation domain of opacity tables

Table 3: Format of OPINTPATH defining the input filenames of the opacity tables

2.2 Installing new OPAL opacity tables

The package allows the use of OPAL tables of any chemical mixture (abundances) for the heavy elements as provided by the OPAL Web-page:

http://www-phys.llnl.gov/Research/OPAL/type1inp.html

After completing the form on this Web-page and pressing the "NORMALIZE" button, you will be asked for your name and e-mail address. You will be notified by e-mail from where you can dowload the new OPAL tables, which will usually have a filename like:

yyyymmdd####.tab

where 'yyyymmdd' represents the date and '####' is a running number (e.g. 0001). This file must be stored as an ASCII file (and not as an HTML file when downloaded with an internet browser like Netscape). The procedure to prepare this new table (actually tables) for the opint package is as follows:

1) go to the main directory of the opacity interpolation package.

2) start the setup progamme

   ./setup.sh

   where you will be asked for the filename (including the complete path) of the newly downloaded opacity table (<table_path>/yyyymmdd####.tab),

   and for the binary format: 'little' or 'big' ENDIAN (little is default). The 'ENDIAN' input is only used for saving the binary tables in the subdirectory 'little_endian' or 'big_endian' located in the directory 'ol95_<name>' (see below). One has still to define the actual binary format by specifying the appropriate compiler option in the Makefile (or it is determined solely by the machine architecture, see also below).

   Next you will be asked for a <name> that will be used to create the new directory 'ol95_<name>', where the new tables will be copied to (e.g. <name> could be a string that indicates the chemical composition (abundances), such as 'AGS05pNe35').

   Finally one has to choose between two (existing) low-temperature AF05 opacity tables of different chemical compositions: AGS04 (default) or GN93, where AGS04 is the Asplund, Grevesse & Noel (2004) composition and GN93 is the Grevesse & Noel (1993) abundances.

3) Edit the Makefile (compiler options, etc.) followed by

   make tab

Once the 'make'-run has (successfully) finished, you will find the appropriate OPINTPATH file for the newly generated tables in the main directory of the opint package.

A similar setup programme exists, **./setup_bigendian.sh**, that allows one to remake the provided opacity tables 2-5 (see Table 1 above) in 'big' ENDIAN format on an appropriate 'big' ENDIAN machine or with appropriate compiler options (e.g. '-convert big_endian' with the Intel Fortran compiler).

3. Using the package

Three subroutines (s/r) have to be called: 1.) **maceps**, 2.) **opinit** and 3.) **opintf** or **opintc** and/or **opints**. The first s/r (**maceps**) evaluates the relative machine precision, s/r **opinit** initializes the opacity-tables i.e., it reads all tables into the machine's memory. The third s/r **opint{cfs}** is(are) the actual interpolation-routine(s) (see below).

```
      implicit double precision (a-h,o-z)
c
c     test-driver program for opacity
c     interpolation subprograms
c     opinit       (opacity initialisation)
c     opintf       (opacity interpolation minimum norm)
c     opints       (opacity interpolation rational splines)
c
      character*80 tabnam
c
      data tabnam /'OPINTPATH'/
      data iorder /4/
      data imode  /2/ ! initialize rat. splines & enables EC
c                     ! use imode /-2/ to disable EC
c
c     get relative machine precision
      call maceps(eps)
c
c     initialize opacity tables
      call opinit(eps,iorder,tabnam,imode)
c
```

Figure 1: Example how to initialize OPINT (excerpt of doptesf.f)

The argument variable **iorder** in call **opinit** defines how many table-points should be used for the univariate interpolation in the X– and Z– domain respectively. This number defines the degree of the used univariate polynomial [Aki91]. Interpolation in Z is performed logarithmical and linear in X. With iorder=4 five tables will be used for the univariate

interpolation in Z and X and will provide the most accurate interpolant (and the most expensive in computation time).

The initialisation subroutine **opinit** has the following arguments:

argument	type	inp/out	description
eps	double precision	input	used to evaluate certain machine-depend tolerances; evaluated from s/r maceps
iorder	integer	input	iorder+1 X and Z table-points will be used for the univariate interpolation in the X and Z domain, respectively $2 \leq$ iorder ≤ 4
tabnam	character*(130)	input	defines the filename, which opacity-tables should be used; take care of the sequence as given below in Table 5
imode	integer	input	selects algorithm(s) to be used; imode=0,1: only the minimum norm algorithm will be initialized (EC is enabled with imode=1) imode=2: only the birational splines are available (EC enabled) imode>2: both algorithm, the minimum norm and birational splines (+EC) are initialized (see also Table 4) imode$\leq$0: a negative value of the particular imode number disables the contributions due to electron conduction

Table 4: argument description of routine opinit

There are three interpolation-routines available, which have to be selected (initialized) properly by the argument variable **imode** in the previous call **opinit** (see also Table 4):

imode	interpolation-routine	description
1	opintf	minimum norm fast version **without** extrapolation-domain checking + EC
1	opintc	minimum norm version **with** extrapolation-domain checking + EC
2	opints	birational splines **with** extrapolation-domain checking + EC
≤0	Disables EC	Electron conduction (EC) is **disabled** with a negative value of the particular imode number (e.g., imode= -1 initializes either opintf or opintc and disables electron conduction

Table 5: available interpolation subroutine calls

The following listing is an excerpt of the programme doptesf.f:

```
c
c    interpolate to get opacity value opalg
c    iexp=0
     ier=0
     call opintf(x,z,tlg,rlg,opalg,opr,opt,opx,opz,iexp,ier)
     if(ier.gt.0)write(6,'(a,i3)')
    +   'opint: ERROR in interpolation s/r, ier=',ier
c    if(iexp.gt.0)print *,'nr. of extrapolated points: ',iexp
c
```

Figure 2: Example how to use routine opintc

For the extrapolation domain checking of the OPAL-tables use **opintc** instead of **opintf** and uncomment:

```
c    iexp=0
c    if(iexp.gt.0)print *,'extrapolated points: ',iexp
```

The OPAL95 tables do not provide values for the region $7.2 \leq \log_{10}(T) \leq 8.7$ and $-0.5 \leq \log_{10}(R) \leq 1.0$. This domain (here called the extrapolation domain) has been reconstructed by a proper extrapolation scheme using the univariate Akima interpolation scheme. The argument variable **iexp** indicates how many of these extrapolated table-points have been

used in the interpolation process. Moreover, in this version the range of $\log_{10}(R)$ has been extended to $\log_{10}(R) = 5$, using linear extrapolation in $\log_{10}(R)$. This will allow an (very) approximate estimate of the opacity for low-mass stars.

The argumentlist is the same for all three routines opintf, opintc and opints:

argument	type	inp/out	description
x	double precision	input	hydrogen mass fraction X 0.0 $\leq$x$\leq$ 0.9
z	double precision	input	mass fraction of heavy elements Z 0.0 $\leq$z$\leq$ 0.1
tlg	double precision	input	logarithm of base 10 of temperature T {3.0}2.70 $\leq$tlg$\leq$ 8.7
rlg	double precision	input	logarithm of base 10 $\frac{\rho}{T6^3}$; ρ = density; $T6 = \frac{T}{10^6}$; $-8. \leq$rlg$\leq 5.$
opalg	double precision	output	logarithm fo base 10 of opacity κ
opr	double precision	output	partial derivative $\frac{\partial(\log\kappa)}{\partial(\log\rho)}_{T,X,Z}$
opt	double precision	output	partial derivative $\frac{\partial(\log\kappa)}{\partial(\log T)}_{\rho,X,Z}$
opx	double precision	output	partial derivative $\frac{\partial(\log\kappa)}{\partial(\log X)}_{\rho,T,Z}$
opz	double precision	output	partial derivative $\frac{\partial(\log\kappa)}{\partial(\log Z)}_{\rho,T,X}$
iexp	integer	output	opint[cs]: # of table points lying in the extrapolation domain opintf : 0
ier	integer	output	if ier = 1, the point lies outside the given table domain, and will be estimated by extrapolation using Shepard's method; otherwise ier=0

Table 6: argument description of routine opintf (opintc, opints). Curly brackets indicate values for AF94 tables.

4. Linking the interpolation routines with the main programme

After successful compilation of the OPINT-routines, the link editor under UNIX may be called as

$(F77) $(FFLAGS) *your_programme* -L*your_path*/v9/lib -lopint

5. Common-blocks

The following common blocks must not be altered between s/r call opinit and s/r opintf (opintc, opints):

```
c
c     pointers for the s/r opintd (used as argument
c     for s/r masubi)
      common /jpoint/ nti
      common /ipoint/ iali
c
c     partial derivatives and triangulation indices
      common /pderiv/ iwk(niwkl),wk(nwkl,ntab,nzva)
c
      common /opadat/ opa(nxir,nyir,ntab,nzva),
     +                rlg(nxir,ntab,nzva),
     +                tlg(nyir,ntab,nzva)
      common /valdat/ ivalo(nyir,ntab,nzva)
      common /tablex/ xtab(ntab),iorder
      common /tablez/ ztab(nzva)
      common /xyzdat/ xd(ndat,nzva),
     +                yd(ndat,nzva),
     +                zd(ndat,ntab,nzva)
      common /machin/ drelpr,toll,eps10
c
c     birational spline coefficients
      common /birasp/ pp,tls(nyif),
     +                ra(nxir,nyif,4,4,ntab,nzva)
c     table dimension for s/r opintcf and opints
      common /tabdim/ nzvai,ntabi,nxiri,nyiri,nyisi,iyifi,mdi,
     +                nti,iali
```

Figure 3: common-blocks used in the OPINT-package

6. File unit numbers

The following logical unit numbers are use in subroutine "opinit" in open-statements:

$lun = 29,\ 31,\ 32,\ 33,$

and should therefore not be used in the calling routines.

References

[Aki70] H. Akima. A new method of interpolation and smooth curve fitting based on local procedures. *J. ACM*, 17(4):589–602, October 1970.

[Aki91] H. Akima. A method of univariate interpolation that has the accuracy of a third-degree polynomial. *ACM Trans. Math. Softw.*, 17(3):341–366, September 1991.

[Kur91] R. L. Kurucz. New opacity calculations. In L. Crivellari et al., editor, *Stellar Atmospheres: Beyond Classical Models*, pages 441–448, Netherlands, 1991. Kluwer Academic Publisher.

[MC89] L. B. Montefusco and G. Casciola. Algorithm 677 C ^{1} surface interpolation. *ACM Trans. Math. Softw.*, 15(4):365–374, December 1989.

[Nie80] G. M. Nielson. Minimum norm interpolation in triangles. *SIAM J. Numer. Anal.*, 17(1):44–62, February 1980.

[Nie83] G. M. Nielson. A method for interpolating scattered data based upon a minimum norm network. *J. Math. Comp.*, 40(161):253–271, January 1983.

[RI92] F.J. Rogers and C.A. Iglesias. Radiative atomic rosseland mean opacity tables. *Ap. J. Suppl.*, 79(2):507–568, April 1992.

[RI95] F.J. Rogers and C.A. Iglesias. In *A.S.P. Conf. Ser.*, Vol. 78, p. 78. Brigham Young Univ., 1995.

[She68] D. Shepard. A two-dimensional interpolation function for irregularly spaced data. In *Proceeding of the* 23^{rd} *National Conference*, pages 517–523, New York, 1968. ACM.

[Spa91] Helmuth Spath. *Zweidimensionale Spline-Interpolations-Algorithmen.* Reihe Datenverarbeitung. R.Oldenbourg Verlag, Munich Vienna, 1991.

Table of contents